Issue 29 | Jun 2010

Blender learning made easy

blenderart MAGAZINE

Industrial Revolution

Creating Blender icons for Adobe Flash

Building a Steampunk Engine

Rigging an Assembly Line Robot Speed

Creating a Forklift

COVERART Steampunk Camera - by Mateusz Walus

EDITOR
Gaurav Nawani gaurav@blenderart.org

MANAGING EDITOR
Sandra Gilbert sandra@blenderart.org

WEBSITE
Nam Pham nam@blenderart.org

DESIGNER
Gaurav, Sandra, Alex

PROOFERS
Brian C. Treacy
Bruce Westfall
Daniel Hand
Daniel Mate
Henriël Veldtmann
Joshua Leung
Joshua Scotton
Kevin Braun
Mark Warren
Noah Summers
Patrick ODonnell
Phillip
Ronan Posnic
Scott Hill
Wade Bick
Valérie Hambert

WRITERS
Jarred de Beer
Dreamsgate
Meino
Alain Mathez
Luiz Eduardo Borges
Randy Blose
Neal Delfeld
apostroph3d

COVER ART
Steampunk Camera - by Mateusz Walus

CONTENTS

EDITORIAL

Sandra Gilbert
Managing Editor

> *"These days a rather large number of people possess gadgets of one sort or another that have no productive purpose other than entertainment "*

Pondering the phrase "Industrial Revolution' brings to mind factories, large machines and the ability to mass produce products or process materials. And of course that certainly did happen, and continues to this day, but like most things in life, I'm betting it started on a much smaller scale. With perhaps a gadget. That sounds kind of silly, but I bet it's true.

Man has shown a remarkable ability to create gadgets to perform mundane repetitive tasks. Most gadgets probably resembled a hodgepodge of gears, levers and pulleys, but they got the job done.

Man has also shown an almost obsessive need to not only make life easier with gadgets, but also more fun. These days a rather large number of people possess gadgets of one sort or another that have no productive purpose other than entertainment.

Obviously today's gadgets do far more than the first ventures into gadgetry did, but the old style gadgets still appeal to us on some level as seen by the popularity of "steampunk".

Steam punk and all its "sub-sister" punk looks, appeal to us at the gut level. Gears and levers bring to mind a simpler time when things made sense and you didn't need an engineering degree just to turn your TV on (okay maybe that is a tad bit exaggerated, but you get the idea).

So as you might have guessed, issue 29 is all about machines and gadgets, and since there is no time like the present, let's get this "Industrial Revolution" started.

IZZY SPEAKS : Gears

> *"I suppose an array modifier could take a lot of the pain away, but I'm still seeing uncomfortable amounts of math in my future."*

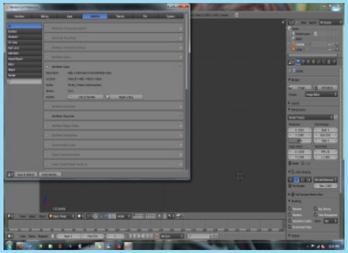
I am not all that mechanically inclined - in real life or artistically (3d modelling) - which so far hasn't really been much of a hardship. While preparing for this issue of Blenderart Magazine, I became very intrigued by all of the steampunk designs I was seeing.

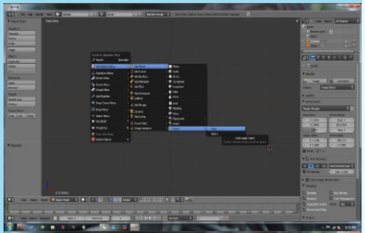
At first glance most of it looked very complicated and very confusing. Upon closer inspection of the simpler images, I realized that a lot of Steampunk designs only looked complicated. For quite a few of them, there was generally a basic shape (the base of the steam punk object) that was decorated with a variety of gears and tubing. Now granted, the more complicated ones that display massive amounts of gears and levers etc, are still beyond my ability or patience to attempt, but these simple ones were actually simple enough even for someone as mechanically challenged as I am.

In fact most of us can model the basic objects and the tubing doesn't look that scary. Gears, on the other hand, look like a right royal pain to model. I suppose an array modifier could take a lot of the pain away, but I'm still seeing uncomfortable amounts of math in my future. Hmm, I don't like math. In fact, I do believe this is in large part why I avoid mechanical modelling to begin with.

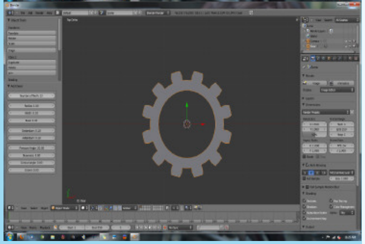
Well luckily for those of us who want to explore a bit of gear making without having a brain melt down, there is a very handy little Add-On that makes gears absolutely painless.

So first up, we need to enable the "Gears" Add-On (File› User Preferences› Add-Ons tab). Scroll down to "Gears" and check the little box on the far right. Enabling the "Gears" Add-On, places a gear mesh object in the "Add Mesh" menu.



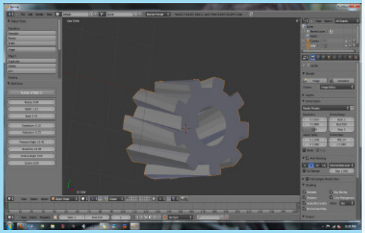


Now adding a gear is as simple as adding any other mesh object.



When the gear is added, options for your gear appear in the Tool Bar menu.

Now I will be the first to admit that most of the options were meaningless to me when they first appeared. But quick experimentation with the different options made it clear what option affected what part of the gear.



So there you go, even a mechanically challenged modeller like me can create a gear in a matter of seconds. Also, for those of you who undertake massive mechanical projects with large amounts of gears, this add-on could prove to be a real time saver.

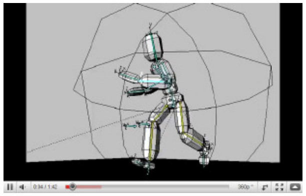
Now I can go tackle a few Steampunk projects as the mood takes me.

Shapeways Material Sample Pack

Ever since Shapeways launched their first beta tests, they have been working hard to improve their 3D printing services. What started out as one material choice has become a wide variety of material options, finishes and colors. And to make it easier to choose the material that best suits your project, they now offer a Material Sample Pack.

Finally you can have a sample of each of the materials Shapeways offers in your hands to touch, scratch, bend and burn (*do not inhale*). This is the perfect way to chose the right material for your next 3D printed application, also a great tool to take to client meetings or trade shows to show potential clients the material range you can offer your/their designs in? Send it as a gift to a maker friend and see how quickly they too become addicted to realizing their designs... Buy it now for $30 (but includes a $25 voucher).

Little Ninja Project



Shane Newville, creator of “Little Ninja Project” has released a video describing his animation techniques. If you want to read more about the Little Ninja Project, download issue #23, Shane wrote a very nice article about creating and producing the “Little Ninja Project”. His video used the blend file supplied with issue #23, to show you how to combine action sequences.

http://www.youtube.com/watch?v=gDZaZVr6uHQ

Contest: Design Your Own Electric Guitar

The
Zoybar Shapeways & Blender Art Magazine Challenge
Enter To Win a Zoybar Guitar With Your Custom Design!
A Collaboration Between Shapeways Zoybar & Blender Art Magazine

We're working with Shapeways and Zoybar on a fun and exciting new contest: Design original hardware features that can be incorporated with the Zoybar instruments and that can be 3D printed with Shapeways.

A platform for Open Hardware, working together with Open Software and an Open Production platform - how cool is that? Read on for contest details.

The Challenge

Design original hardware features that can be incorporated with the Zoybar instruments and enter a contest to win your creation. Design in Blender.

How does it work?

- Zoybar offers a number of CAD files that you can base your work on. Download them and design your new instrument.

- Upload your design to Zoybar to participate in the contest (tip: upload to Shapeways too, to test the printability of your design. Tag with 'zoybar' please).
- Check out this diagram to learn how to participate and for all the relevant links.
- Timeline: The competition will start on August 05, however submissions will not be revealed before August 22 and will continue until September 20, 2010.The competition will end on October 05, 2010.

The prizes

Function category prize (Open to all): A Full Zoybar Guitar or Bass Hardware kit provided by Zoybar, plus 3D print of the winning design entry in the amount of $200 (USD) provided by Shapeways. Total Value: $870 (USD)

Fantasy category prize: (Exclusive for Blender users only) A Full Zoybar Guitar or Bass Hardware kit. Valued: $670 (USD) provided by Zoybar.

The winners will be announced in Blenderart Magazine's 30th issue and on the Shapeways blog.

Course: 3D-World with Blender

A new blender study/support-group has started in Tampere, Finland.

Scaia, a Finnish course based school for digital arts, have opened their new internet pages and the courselist for September.

Scaia teaches with software that have a free/semifree alternative for anyone who wants to express themselves with 3D-art, digital painting, blogs or even with making games. Blender is definitely Scaia's favorite software, but in September it will also be possible to learn Gimp, Unity or Wordpress.

Scaia is also planning to release one or more online tutorials about Blender during the course semester. Here is a little teaser what might be ahead: http://vimeo.com/13055184

Scaia is powered by the help of TAMK, Tampere University of Applied Science - Art and Media, and a student based co-operative, High Score.

Most of Scaia's talented teachers are students or ex-students of TAMK.

Links:

Frontpage, Right now only in Finnish :(

Course: 3D-World with Blender, there are two video-clips about the upcoming course. Bottom one is a teachers example rendered from a previous Blender course early 2010.

http://www.scaia-arts.com/blog/?page_id=34

Proto, Scaia's mascot android, is made with Blender and Photoshop, Artist: Miikka Harjuntausta.

Christmas Co-Creator Contest 2010

After the great success of last years Co-Create Christmas Contest, we have to decided to run the contest again.

Once again we are looking for designs that embrace Co-Creation by allowing consumers to personalize the design whether it is for themselves, or as a gift for others. Take a look at the Christmas Page to see some of the creative Christmas designs the Shapeways community have already produced.

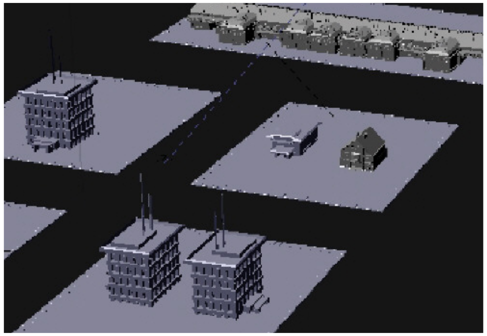
Prizes include:

- Top five finalists receive a free 3D print of their design and be included in our Christmas promotions,
- Winner to also receive a Shapeways Voucher for $100*.

To enter:

- Submit your new Co-Creator with the tag "Xmas2010" by the 30th of September 2010.
- Product can not cost more than $100 (excl. VAT and mark-up) and needs to be Christmas themed.
- The entry must be uploaded to Shapeways, be public and a Co-Creator on the Christmas page.
- The winner and runners up will be notified in writing by 11th October 2010.
- No purchase necessary
- By entering this competition, entrants will be deemed to have accepted and agreed to the conditions.
- No cash or other alternative prizes available.
- coupon valid till 24 December 2010, only 1 coupon can be used per order.
- The prize draw is not open to Shapeways employees or their families.
- The promoter's decision is final – in case of dispute, no correspondence will be entered into.
- Winning entries will be documented by Shapeways and may be used for promotional purposes.
- Winners retain all IP as per Shapeways standard terms and conditions
- Winners and winning entries may be required to take part in any publicity resulting from this competition.
- Promoter: Shapeways, High Tech Campus 27, Eindhoven 5656 AE, The Netherlands.

Visit the blogpost for comments and questions.



Blender icons for Adobe Flash

Neal Delfeld

Introduction

This tutorial shows the process for producing isometric icons for use in Flash. Objects are brought into Blender; Blender is used to produce both a backdrop image and an icon sheet; Photoshop processes the icons and produces icon images with an onscreen position as a name; and Flash completes the picture.

Overview

In a demo Adobe Flash game for a client, we were having a difficulty keeping costs down. Much of this came down to the fact that if you go over 250,000 viewers, the prices from many stock image companies quadruples (also annoying was that some stock image sites do not allow video game usage).

Before discovering these problems we looked at purchasing isometric images. The isometric point of view is consistent between different artists' products (if truly isometric). And in theory we could add our own images in some situations for less overall costs.

This is especially true in terms of creating background scenery and geometric objects, like buildings. Since there are a number of free building and scenery models for Blender, I really only needed the process for producing isometric images.

I should note that the demo showed we would not be able to meet the client's pricing, so this isometric process has some flaws that would have been corrected if we had been able to work on the project.

Setting up the scene

Create a null object, and make sure it is at the origin with no rotation. Name it "origNull".

Add a single light source and adjust to taste. I find it produces pretty good shadows without cluttering up the scene with extra patterns.

Add your scenery! This part is up to you, but I used a city metaphor for this demo. Divide your artwork into appropriate layers. The buildings which I wanted to turn into icons are on a single layer. The main scenery will be black and white, so most objects are on a single separate layer. I wanted the street to be colored so I put that on it's own layer. Actually, if you want to preserve the coloring for an object, it can go on this layer as well since the compositor set up will not remove colors from this layer.

Adjusting the camera.

Select the camera. Switch to the Constraints panel (Object (F7) ›› Constraints).Click "Add Constraint" and choose type "Track To". In the "Target" text, type the name of the origin null object ("origNull"). The "To:" button setting should have only "-Z" active, and the "Up:" button setting should have the "Y" active. Move the camera to an "isometric" position. I used fake isometric since it was easier to position the camera, mine is just the point where the distance along the axes are all the same. I have my camera at (200,200,200).

To help make an object look isometric, all objects should treat the origin axes as their direction axes. To make a true isometric layout read Tom Gersic's great blog post titled "Rendering Isometric Tiles in Blender 3D".

Neal Delfeld

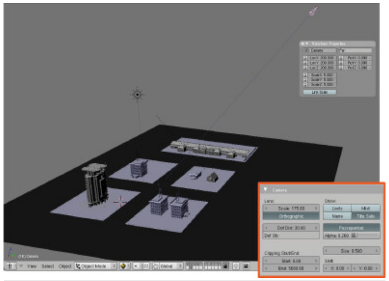


Figure . Camera settings.

Select the camera and change to the orthographic view (Editing >> Camera, click the "Orthographic" button). Now is a good time to set the camera clipping. This is the distance at which the camera no longer views the scene. Clipping should be large enough to view the entire scene no matter how far away. I set mine to 5000.

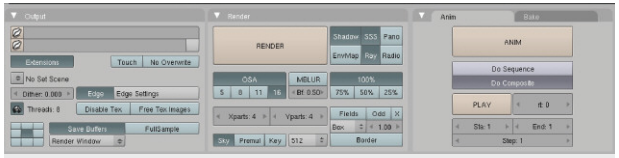
Creating "Render Layers".

Create a series of Render Layers (Scene (F10) >> Render Layers). Under the "Scene:" arrows, select "ADD NEW". Name the new layers: "Edges", "Shadows", "buildings", and "street" (Note that "buildings" and "street" render layers might be better labeled "icons" and "preserveColor" respectively).

Make a viewport for the Node Editor. Add one "Render Layer" node for each render layer. Add a "Split Viewer" node. Render the image to load the buffers.

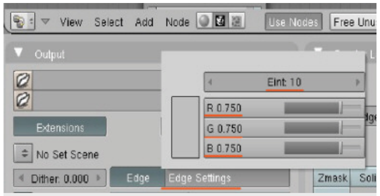
Z-masking has to be exported on some of the Render Layers since the compositor will act on everything visible. Z-masking will hide things behind other things. For example z-mask can hide (not render) the shadows that are hidden behind buildings. The compositor will only work with the shadows which are visible in the final render.

Render Layers:



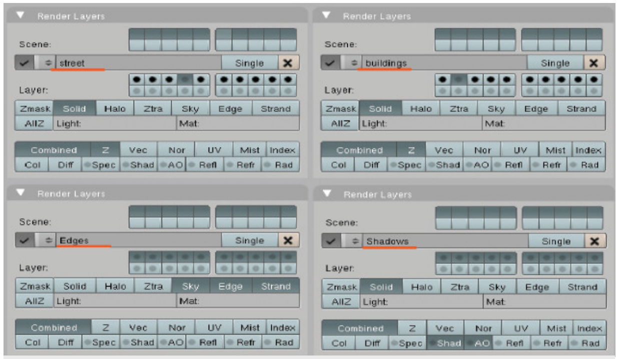
Dither: 0.000
Threads: 8
Xparts: 4
Yparts: 4
Bf: 0.50
Box 1.00
512
Render Window
fr: 0
Sta: 1
End: 1
Step: 1

Figure . General Render Settings.



Eint: 10
R 0.750
G 0.750
B 0.750
Dither: 0.000

Figure . Edge settings.



Scene: street Single
Scene: buildings Single
Scene: Edges Single
Scene: Shadows Single

Figure. Four Individual Layer Settings.

Neal Delfeld

I have divided my scene into four concepts for the compositor:

1. Buildings (icons)

All the buildings that I want to turn into icons are kept on one layer. The compositor will "tint" these buildings. I could have colored them individually, but I like the results of the compositing process better. So the compositor makes this layer black-and-white, tints it a color and adds it back into the scene.

2. Streets

This layer preserves the texture colors of the objects as they are. I have this on a separate layer since I wanted to create a specific effect with the color. (In the instructions that follow I have at times changed the color to bright red for the object on this layer in order to show what is going on the red texture is not the final color, though.)

3. Edges

This layer exists to deliver the edge effect that makes the scene so cartoonish. This effect was drawn mostly from this great tutorial: Please read it! To summarize what techniques I used:

In the Render ›› Output panel:Make the "Edge" button active

Click on "Edge Settings", and adjust the color and intensity. In the Render ›› "Render Layers" panel:

Select all the object render layers (not the light source layer). These are in the second set of layer boxes (the ones with the dot in the layer button) and are not the same as the 3D layers.

Only these four buttons should be active: "Sky", "Edge", "Strand", and "Combine".

4. Shadows

This layer controls shadows and ambient occlusion effects. If these are too strong the scene appears cluttered. So they are softened in a number of ways. The origin for these effects is again from.

Render Output

This node setup outputs 2 images: the overall image, and the icons with an alpha channel.

Compositing Process

For most of the following images, I am showing output from the SplitViewer node. I don't see a lot of tutorials that use SplitViewer but I have found it invaluable for testing compositor effects.

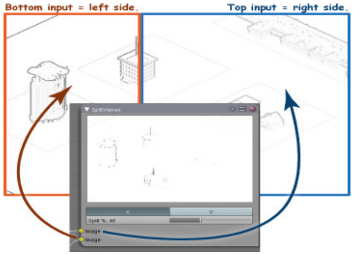


Figure . How the Split Viewer node works.

The following set of images go into detail about the compositor set up.

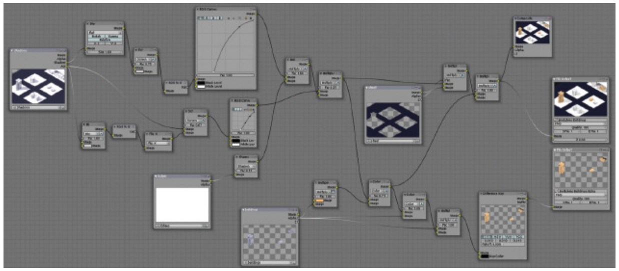
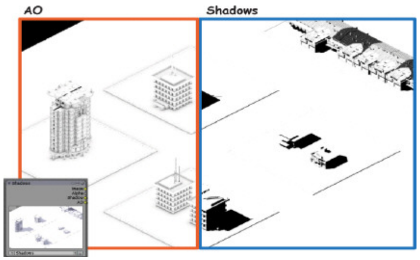


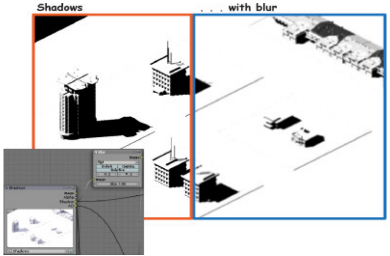
Figure . Node Editor (compositor) setup for this demo.



AO

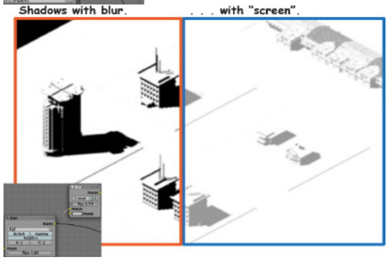
Shadows

Figure . Initial AO & Shadow layer



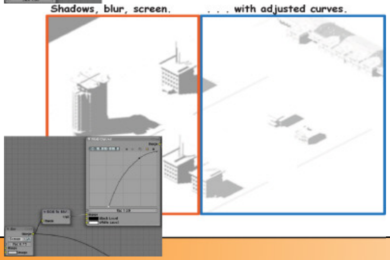
Shadows

. . . with blur



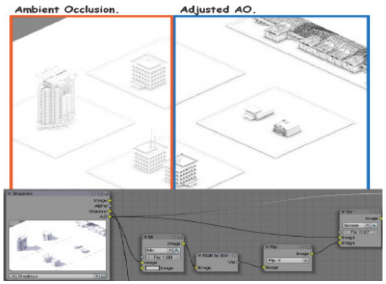
Shadows with blur.

. . . with "screen".



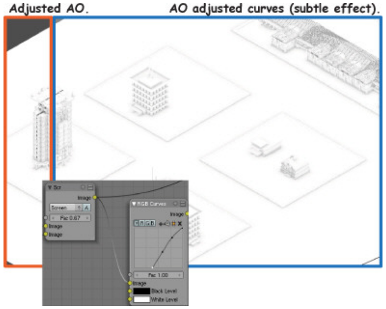
Shadows, blur, screen.

. . . with adjusted curves.



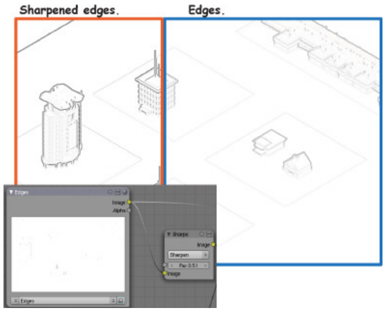
Ambient Occlusion.

Adjusted AO.



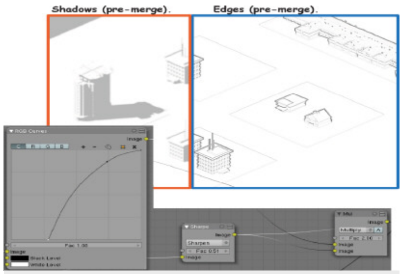
Adjusted AO.

AO adjusted curves (subtle effect).



Sharpened edges.

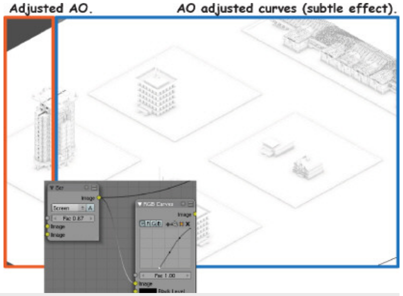
Edges.



Shadows (pre-merge).

Edges (pre-merge).

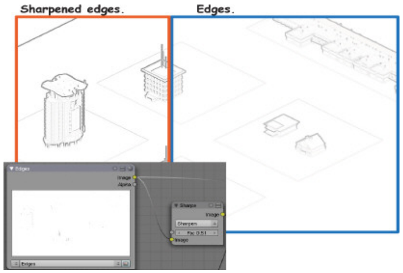
Figure . Before merging shadow with edges.



Adjusted AO.

AO adjusted curves (subtle effect).

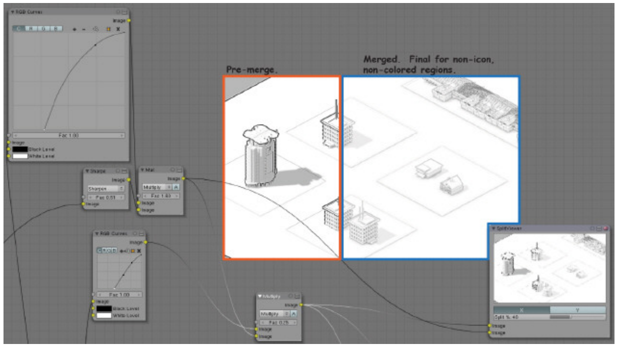
Figure . After merging shadow & edges.

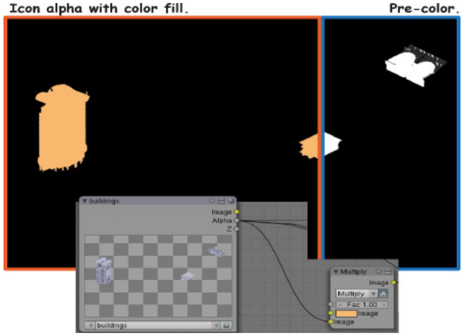


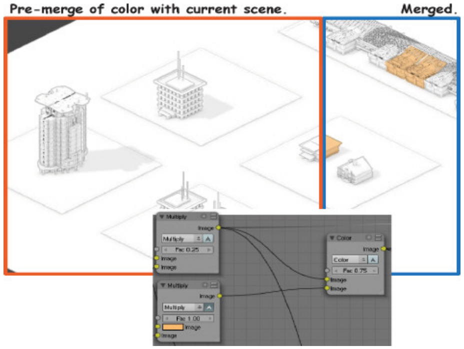
Sharpened edges.

Edges.

Neal Delfeld







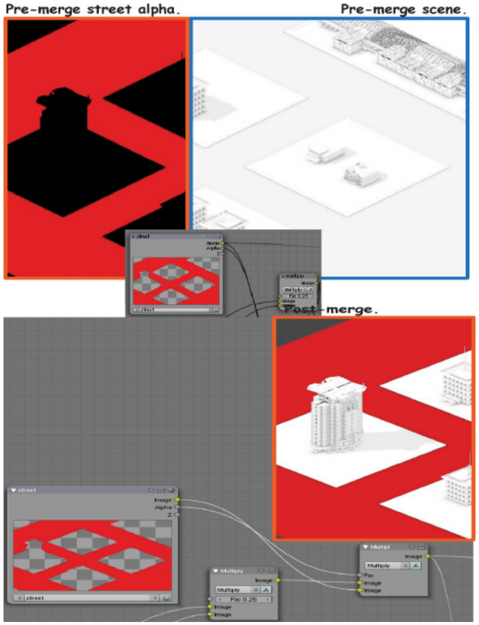


Figure . Final Output No.1.

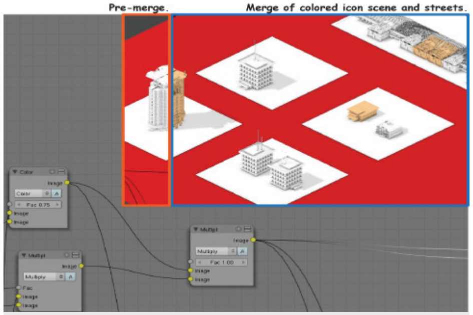


Figure . Final Output No.2.

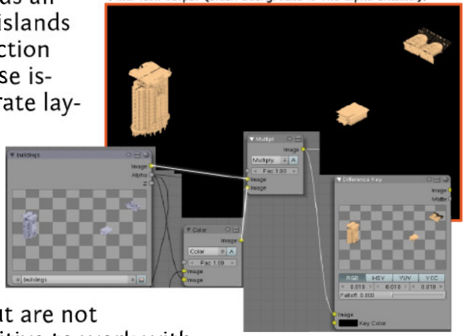
Neal Delfeld

Neal Delfeld

Photoshop

NDSplitSelectionRegionsToLayers.jsx Description

Final icon output (black background is the alpha channel).



This script finds all the selection islands within a selection and saves those islands as separate layers. The islands are actually stored as an array within the PS scripting environment, but are not the most intuitive to work with. One by one the islands have to be converted to a PS path and the path has to be converted to a (separate) selection. Each path knows it's position onscreen by referencing the top-left corner point. I use this point to name the layer (later, this naming convention allows me to place the icon in the same position within Flash).

I would like to be able to follow the same techniques in Gimp, but I do not know how to programmatically find the selection "islands" in Gimp. Even in Photoshop, in order to make this an automated process, I had to write a PS script. Also note that I do not know Python.

It may be entirely possible to replicate this PS script entirely in Blender.

Using NDSplitSelectionRegionsToLayers.jsx

Download this script from herer:

- The blog post is password protected. The password is: *blenderart*
- Save the script as a text file named: *NDSplitSelectionRegionsToLayers.jsx*
- Bring the building icon render into Photoshop. It's probably best to make a new PSD.
- Use the icon layer by highlighting it in the Layers palette if it's not already. Using the "magic wand" selection tool, set the "Tolerance" set to 0, and make sure that "Sample All Layers" is unchecked. Select the non-icon area within the icon layer. This will create a selection of everything except the building icons and that's what you want since the script will invert the selection.
- Run NDSplitSelectionRegionsToLayers.jsx. (File ›› Scripts ›› Browse . . ., and locate the script). There may be some pop-ups that will indicate coordinates for each island found. These can be stopped by commenting out the line:alert(myPathsArray[s + 1]);

After the script is complete you should have one layer for each icon plus the original layer.

In the Layers palette, make the original layer invisible (click the "eye" next to the layer name to hide it).

Using the Photoshop Script "Export Layers To Files"

Run the Photoshop script: File ›› Scripts ›› "Export Layers To Files"

In the popup, check "Visible Layers Only", "Transparency" and "Trim Layers".

Make sure you output a transparency type, like PNG-24. Add a prefix name, choose an output directory, and hit "Run".

The output should be small icon files, one icon per file.

Flash

Bring both the main image and all the individual icon images into Flash.

If you allow Flash to automatically create MovieClips (MCs) out of each of the icon images, the name of the MCs should be the file names. If you have MCs that do not match their image names, rename them to match.

Position each icon at the XY position where it's name indicates. That's it! (This is scriptable, but I did not have the opportunity to create it - sorry!).

Now you can assign controls to these icons. . . try changing the filter on a MouseOver event to highlight the icon. The transparent areas are not highlighted, so this is a pretty nice effect.

Conclusion

I hope this helps you get better icons in your Flash games with more consistency and less cost. Happy coding!

Works cited

Edge shading, de-coloring: http://matthieu3d.free.fr/TutoVira1/tuten.html

Isometric projection: http://www.gersic.com/blog.php?id=52

Docs on compositing: http://www.blender.org/development/release-logs/blender-242/blender-composite-nodes/

Docs on Photoshop Scripting (CS3, CS4, CS5): http://www.adobe.com/devnet/photoshop/scripting/

Neal Delfeld

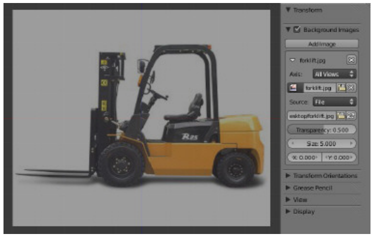
Creating a Forklift

Jarred de Beer

Introduction

The first step was to get reference images from a google image search. If you do a google image search for 'forklift', the second image result is rather funny. However, I ended up going with this one: because it was profiled nicely.

I then set it up as a background image in Blender. To do this, enable the 'Background Images' tab by clicking on the tick box. Open up the tab and click on 'Open', browse to your image and select it. If you move into a profile view (front, side, top etc) you'll see the background image. Tweak the settings according to your liking.



▼ Transform
▼ ☑ Background Images
Add Image
forklift.jpg
Axis: All Views
forklift.jpg
Source: File
esktop/forklift.jpg
Transparency: 0.500
Size: 5.000
X: 0.000 Y: 0.000
▶ Transform Orientations
▶ Grease Pencil
▶ View
▶ Display

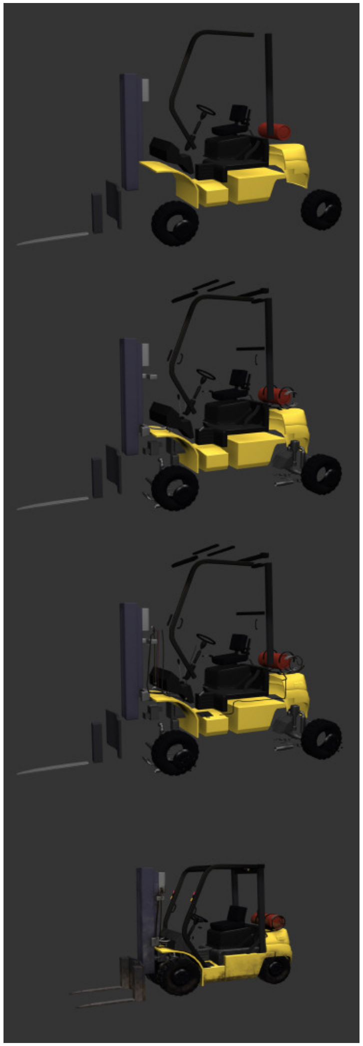
MODELLING in levels of detail:

The modelling was broken up into three levels: Primary detail, Secondary detail and Tertiary detail. It might be advisable to keep your modelling organised by placing them on different layers. e.g. layer1, 2 and 3.

NOTE: The objects in the illustrations below are separated for visual purposes only. Don't model them this way, model them according to the reference image.

Primary detail consisted of main forms, such as the body, wheels, cage, fork lift, tank, seat and wheel. These are essentially identified from the forklift's structure, and are responsible for its silhouette.

Because it's an engineered device, it's symmetric, so I only modelled the left half and used a mirror modifier on the objects to mirror them. For the wheels you only need to model one quarter of its profile, and then use the mirror modifier twice to complete it.



Jarred de Beer

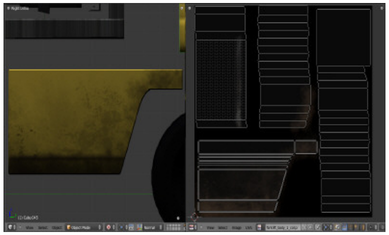
Large forms were broken up into smaller parts, such as the panels of the body, the blocks of the fork mechanism etc. This keeps it more believable and helps with placing secondary and tertiary detail.

Secondary detail consisted mainly of smaller functional items, such as levers, pedals, lights, electrical boxes, engine pieces etc. This helps to make it appear functional. For these you don't have to stick as rigorously to the reference image and can start looking at other images for inspiration. You also don't have to continually reference it in side view to see if it matches.

Tertiary detail consisted mainly of smaller parts to help pull it all together and keep the eye busy. Namely: cables, an air vent, wheel nuts, shocks and wiring.

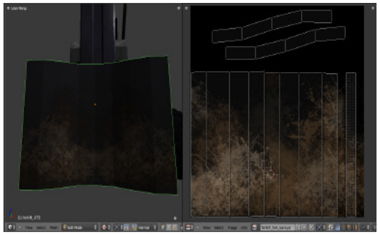
Putting them all together, and throwing in some texturing, we have our forklift:

TEXTURING:

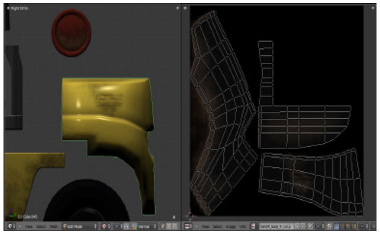


Unwrapping objects is mostly a case of placing your seams and unwrapping by selecting all the vertices, pressing 'U' and choosing the most appropriate unwrapping algorithm. This for *ME* is generally a painful process, but it's very rewarding once you see the texturing results.

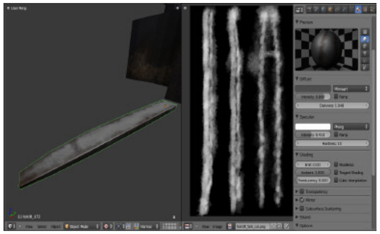
The main aim of texturing was to show that the forklift had been used. This resulted in the strategic placement of dirt and some wear, using some dirt gimp brushes.



I wanted to create the illusion that the forklift had been through puddles of muddied water, which had been kicked up to dirty the body's side. This would also have muddied the forks and the wheel tread. Some dust was given to the tank at the back to indicate that it's a fixture and had been sitting there for a while, and some dirt was applied to the air vent on the back to indicate that the engine's air wasn't always clean.



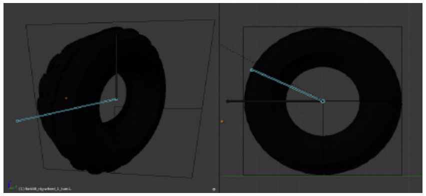
Additional texturing could have been added, but time had to be used decisively according to the deadline. The image below shows that the texture was used as an alpha map on a brown colour, which then comes across over the material colour as if it was dirt.



Some images were sourced from the internet for the metal textures, and procedural cloud textures were used to break up the spec values wherever they occurred. The colour images above were also used to subtly affect bump and spec values.

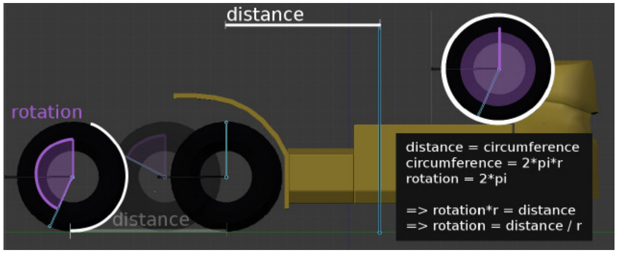
Jarred de Beer

Rigging:

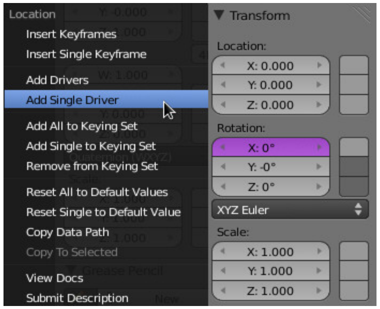


The rigging required that the wheels would be driven, and could turn left and right. A bone was used to manipulate the wheel left and right, and another one to drive it. The driving bone was parented to the left and right bone so that the wheel would still drive properly if turned.

The equation to turn a wheel is actually somewhat simple. The main controller is used to move the forklift around, so the location value of the main controller will give us the distance travelled, which will then drive the wheels rotation. If main controller is moved a distance equal to the circumference of the wheel, the wheel would have rotated a full revolution (in radians: 2*pi). We know that the circumference of a circle is 2*pi*r (where r is the radius of the circle). If the circumference is 2*pi*r and the rotation is 2*pi, then rotation*r is the circumference, which happens to be the distance. So we have: rotation*r = distance. Therefore rotation = distance / r.



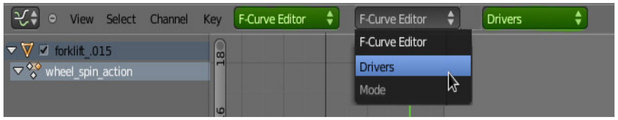
As a result we can use a driver, which takes the location value from the main controller and just divide it by the radius of the wheel. OR, we can create a straight line curve whose slope is the same as the ratio of distance / r and let this drive the wheel. Below I show the former method.



Location:
X: 0.000
Y: 0.000
Z: 0.000
Rotation:
X: 0°
Y: -0°
Z: 0°
XYZ Euler
Scale:
X: 1.000
Y: 1.000
Z: 1.000

With the rolling wheel bone selected, in the transform properties of the 3D View, right click on 'Rotation X' and click 'Add Single Driver'. The property will turn pink indicating that a driver has been set on it.

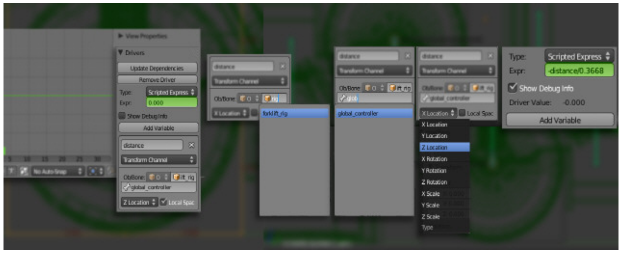
To navigate to the drivers, press F6 in a window to bring up the f-curve editor. Then click on the 'Mode' drop down box, which will say 'F-Curve Editor' and select 'Drivers'.



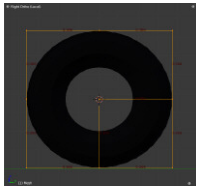
This will bring you to the Drivers window. From here, on the right side of the window near the bottom is a 'Modifiers' tab, remove the default 'Generator' modifier (NOTE: This is where you can set the slope to be the same ratio as distance / radius, as mentioned above. This is easier than setting the drivers as below, but if you're curious you can follow on).

Jarred de Beer

Click on 'Add Variable' in the Drivers tab. Rename the variable to distance. Change the type from 'Single Property' to 'Transform Channel' (NOTE: This step is completed -but ignored- in the illustration below). Select the armature object from the 'Ob/Bone' property (In this case forklift_rig). Select the bone responsible for driving the wheel (In this case global_controller). Change the default 'X Location' property to 'Z Location' (In our case +Z Location is forward, this depends on your bones' alignment). Type the expression 'distance / 0.3668' in the 'Expr:' channel

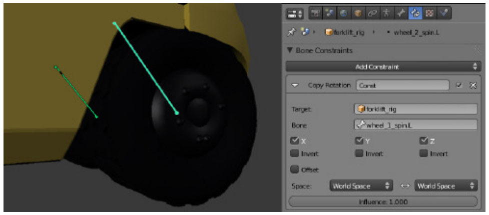

Drivers
Type: Scripted Express
Expr: 0.000
distance
Transform Channel
Ob/Bone: forklift_rig
global_controller
Z Location ☑ Local Spac
Type: Scripted Express
Expr: -distance/0.3668
☑ Show Debug Info
Driver Value: -0.000

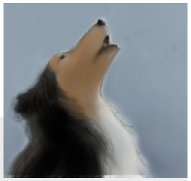
(in our case the radius is 0.3668).



Now when you move the main controller along the Z Location axis, the wheel will turn appropriately. As a further step I created a bone for each wheel, to turn them in the same way as the wheel above.

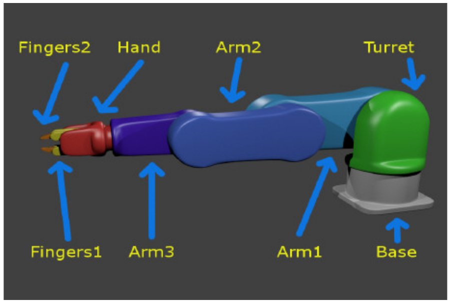
These other three will have 'Copy Rotation' constraints on them, copying the rotation from the bone with the driver on it. This way all the wheels will roll together.


forklift_rig › wheel_2_spin.L
Bone Constraints
Copy Rotation: Const
Target: forklift_rig
Bone: wheel_1_spin.L
☑ X ☑ Y ☑ Z
□ Invert □ Invert □ Invert
□ Offset
Space: World Space ↔ World Space
Influence: 1.000



Jarred de Beer

Jarred de Beer graduated from animation in 2008/09. After interning on Bassam's animation short 'Tube', he's now furthering his animation interests at raramuridesign, traditionally a web development company.



Rigging an Assembly Line Robot

Introduction

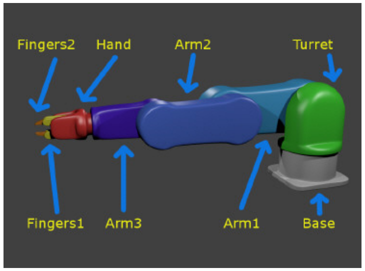
The assembly line is perhaps the most important development in the manufacturing process and the Industrial Revolution. This manufacturing technique enabled increased productivity at reduced costs, so much so that by 1912 a Ford Model-T could be purchased for $575. During the last half of the 20th century the introduction of the assembly line robot also increased productivity and further reduced costs.

In this article I will be covering a few important aspects of mechanical rigging in Blender, then we will dive into rigging an assembly line robot with an armature for animation. While I wrote this article using Blender 2.5 alpha 2, the same techniques can be used with Blender 2.49b as well.

Before we dive into this project, let's cover a few basics I like to adhere to for mechanical rigging. First off, it's very important that the root and tip of our armature bones are positioned exactly at the center of the rotation of the moving parts. If we don't do this the object will appear to be falling apart or broken when animating. To accomplish this we will make extensive use of snapping (Shift-S on the keyboard). I also prefer to keep all moving parts as separate mesh objects.

This will allow us to parent the meshes to the bones responsible for moving them and avoid the need for weight painting/vertex groups. And lastly, before starting to rig any sort of object we need to ensure that the mesh(es) don't have any scale or rotations to them. To accomplish this, select the mesh(es) in object mode and press Ctrl-A, selecting Rotation from the pop-up menu. Repeat this step again choosing Scale. This ensures that any scale or rotation that the mesh(es) may have acquired during the modeling process is zeroed out. This should always be the first step in the rigging process, failure to do this step can produce strange results further down the line.

Randy Blose



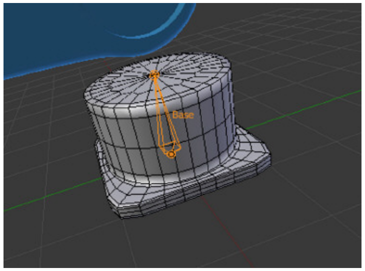
As I created each mesh, I named them to make it easy to understand what each part is. So let's go through and look at the different parts of the robotic arm. In gray is the Base, in green is the Turret, in the varying shades of blue are Arm1, Arm2, and Arm3. In red is the Hand, in yellow is Finger1 and in orange is Finger2 (I named the upper fingers TopFinger1 and TopFinger2, the bottom fingers BotFinger1 and BotFinger2).

Since the Base mesh will be stationary, this is a good place to start creating our armature. Select the central bottom vertex of the Base mesh in edit mode and use Shift-S -> Snap Cursor to Selection. Now, in Object mode, from the add menu select Armature (in 2.5 chose the 'single bone' option) to create our first bone. We'll name this bone 'Base', tab into edit mode and bring the tip of the bone to the top center of the Base mesh.

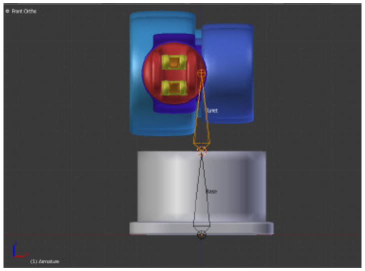
An easy way to do this is to select the Base mesh, enter edit mode, select the top center vertex and snap the cursor to the selection. Then in the armature's edit mode, select the tip of the bone and snap it to the cursor. Remember, we can hide mesh objects in object mode with the H-key, and unhide them with Alt-H to better see what we are doing.

Randy Blose

Now, in the armature's edit mode, grab the tip of the bone we just created and extrude (E-Key) along the Z-axis (Z-Key). It doesn't matter how much we extrude the bone, as we will now position the tip by snapping. Select the Arm1 mesh, enter edit mode and select the vertex at the center pivot point of this mesh where it rotates around the Turret mesh. In the armature's edit mode, select the tip of the new bone and snap it to the cursor.

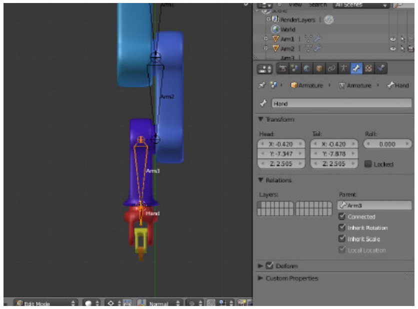


Now, since our objects are offset along the X-axis but the main parts form a straight line along the X-axis, we can position the armature to run straight along the X-axis. The way we snapped the tip of the new bone has caused this bone to be angled along the X-axis. To fix this, snap the cursor to the root of the new bone, set the pivot center for rotation/scaling to the 3d cursor, select the tip of the bone and scale along the X-axis by entering 0 on the keyboard, (S-Key, X-key, 0-Key). The end result should be a perfectly vertical bone when viewed from the front. Don't forget to name the bone!



Extrude 2 more bones along the Y-axis using the techniques just mentioned for the Arm1 and Arm2 meshes.

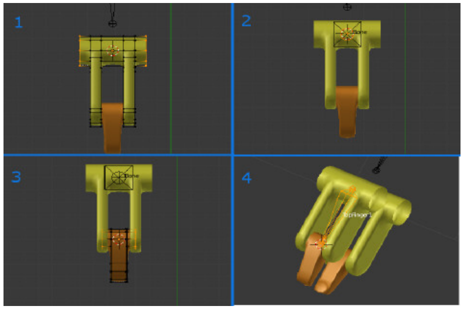
Extrude the tip, snap to pivot point and correct X-axis alignment as needed (if you extrude in the top or side views the X-axis alignment won't be a problem). Repeat for the next bone. Since the Hand will rotate around the center point of the tip of Arm3, the bone for this arm segment should run along the center of this mesh's X and Z axis. To achieve this, I selected the vertices of Arm3 where it will pivot with Arm2, snapped the cursor to selected and added a bone. The tip of this new bone I snapped to the center pivot point of the Hand/Arm3 meshes. Name the bone Arm3, and make it an unconnected child of Arm2 bone. Extrude the tip of Arm3 bone along the Y-axis (E-Key, Y-Key) a bit to make the Hand bone.


Hand

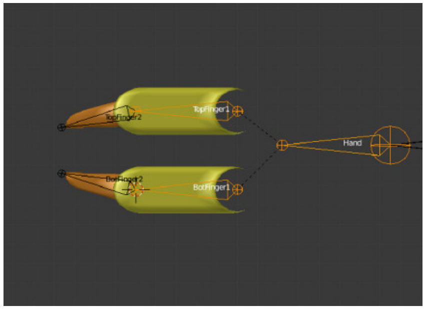
Head: X: -0.420, Y: -7.347, Z: 2.505
Tail: X: 0.420, Y: -7.878, Z: 2.505
Roll: 0.000
Parent: Arm3
☑ Connected
☑ Inherit Rotation
☑ Inherit Scale

On to the fingers! This part was a bit tedious. To make it easier I hid (H-Key) the Hand and Arm3 meshes so I could really zoom in on the finger meshes and concentrate on them. First, I selected the TopFinger1 mesh, entered edit mode, selected the outside two rings of the mesh and snapped the cursor to selection. With the cursor positioned, enter the armature's edit mode and add a bone (Shift-A). Now select the TopFinger2 mesh and in edit mode select the outside two rings, snap the cursor to selection. Select the newly created bone's tip and snap selection to cursor. The new bone is positioned so its root is at the pivot point of the TopFinger1 mesh and its tip at the pivot point between TopFinger1 and TopFinger2 meshes.

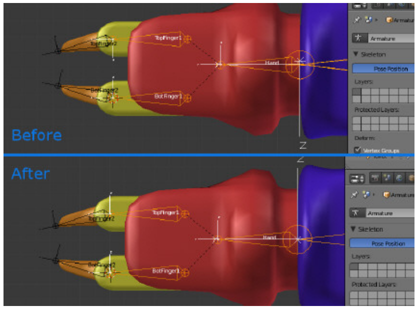
Randy Blose



Extrude the tip of TopFinger1 bone to make the TopFinger2 bone. I extruded mine from the side view and positioned its tip so it was protruding through the TopFinger2 mesh a bit. Extruding in side view ensures the bones will be aligned along the Y-axis correctly. Now repeat this process to create bones for the BotFinger1 and BotFinger2 meshes. The final step here will be to make TopFinger1 and BotFinger1 the children of the Hand bone. An easy way to do this is select the TopFinger1 and BotFinger1 bones in edit mode, then select the Hand bone, press Ctrl-P and select 'Keep Offset' from the pop-up menu.



We need to make an adjustment to these bones to ensure they rotate and move correctly while animating. If you look closely at the bones, they may not appear to be perfectly square and aligned with each other, some may appear slightly twisted. To fix this, we need to correct the bone roll angles. I feel this is an often over-looked but very important step in the rigging process. To see what we are doing a bit better, in the armature panel under display, turn on 'Axes'. This displays the bones X,Y, and Z-axes. Select all the bones in edit mode and press Ctrl-N, selecting 'Z-Axis Up' from the pop-up menu. This corrects the bone's roll angle and aligns them so they all match.



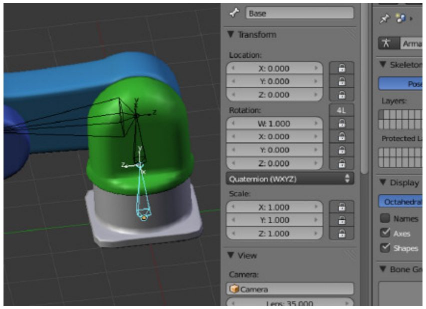
OK, so let's test out the rig. Enter pose mode, select a bone and move it around. Now you might be saying to yourself, "hey, the mesh doesn't move with the bones" and you are correct, the mesh doesn't move. In order to get the mesh to move with the bone, we will parent the different parts of the mesh to the bone that will control it. In object mode, select the base mesh, then shift-select the armature, enter pose mode, select the base bone and press CTRL-P, from the pop-up menu select Bone.

Now the mesh is parented to the bone and moving the bone in pose mode will move the mesh. Repeat this process for all the meshes and bones. The final step is to lock down the location, rotation, and scaling of bones so the bones can't be posed in a way that will make the robot arm appear broken.

Randy Blose

To start, open the Transform Properties window/panel by pressing the N-key from the 3d window. Also, to better see what we are doing, enable axes display for the armature in the Armature Panel. Let's start with the Base bone. Select the bone in pose mode and in the Transform Properties panel, click on all the lock icons by the Location, Rotation, and Scale fields. This will disable the user's ability to move, rotate, or scale this bone.

Why did we do this? Because the base of the robot arm is stationary and not meant to move during animation. The armature can be moved around in object mode to position it within a scene, but in real life, industrial machines like this are anchored to the floor and never move.


Base

Transform

Location:
X: 0.000
Y: 0.000
Z: 0.000

Rotation: 4L
W: 1.000
X: 0.000
Y: 0.000
Z: 0.000

Quaternion (WXYZ)

Scale:
X: 1.000
Y: 1.000
Z: 1.000

View

Camera:
Camera
Lens: 35.000

So let's look at the Turret bone. This bone we want to lock down all the location and scale values. Looking closer at the bone, we can see its Y-axis is pointing straight up and we do want to allow it to rotate on its Y-axis, so we will just lock down the X and Z-Axes for it. Please note that by default in Blender 2.5 Alpha 2 we have 4 rotation fields, the standard X, Y, and Z, plus an additional field for W.

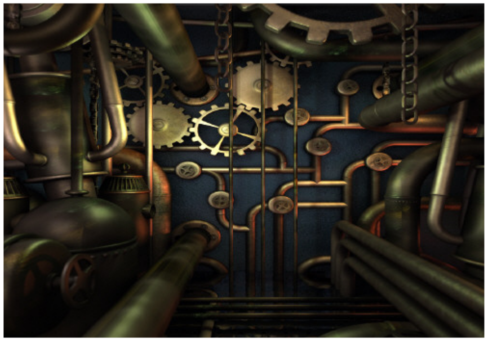
Basically the W rotation is for Quaternion Rotations, but all we really need to know now is do not lock this one. The Arm1 bone we want to allow to rotate on its X-Axis, so lock down everything but the X and W rotations. Repeat this for Arm2 and Arm3. Now the Hand bone we want to allow rotation on the Y axis, so lock down everything but the Y and W rotations. For TopFinger1, TopFinger2, BotFinger1, and BotFinger2 we can lock down all location and scale fields, and lock down the Y and Z rotations.

Now with all those settings made, no matter what bone is selected and how you attempt to move it, it will only move in the way we allow it to. This is pretty much a foolproof rig now and no matter how hard anyone tries, they won't be able to break our robotic arm, unless they change the settings we made.

Conclusion

I hope everyone had fun and learned something from this tutorial! The robotic arm used in this tutorial can be downloaded from here and it is rigged as presented. In addition to writing this article, I've recorded a video demo of the steps presented here, you can view it here. I've also recorded two other videos on this subject, the first one covers additions to this rig resulting in fewer animation controls for easier animation, it can be viewed here. The second video covers tidying up the rig and creating custom shapes for the controller bones, it can be seen here . Good luck and Happy Blending!!!

Randy Blose aka revolt_randy.



Building a Steampunk Engine

Luiz Eduardo Borges

Introduction

With the Steampunk aesthetic in mind, the premise of the work is to create a scene with a half realistic / half cartoony look, similar to those found in many video games. The model represents the engine room of a retro submarine, similar to the watercraft commanded by Captain Nemo from Jules Verne's work.

The Concept

What if computers had appeared in Victorian England? And what about robots? What could the world have become? It seems that these questions have already settled in the minds of many people with interesting ideas.

Inside the book The Difference Engine by William Gibson and Bruce Sterling, the term is exactly that: the computer was built from mechanical (not electronic) pieces during the nineteenth century in England. For some people, this was ground zero of Steampunk literature.

At first, Steampunk was a literary genre, a sort of old fashioned science fiction dominated by technological and historical speculation, usually associated with the nineteenth century and steam power. Based primarily on works of authors such as Jules Verne, H. G. Wells and Mary Shelley, the genre has many features in common with the Cyberpunk and Gothic genres.

The Steampunk genre influenced other kinds of art and became an imaginary complex with a clear philosophy containing concepts such as the enhancement of technology and mastery of it. This is usually represented by the inventors / technicians who, in many aspects, are quite similar to hackers (as they are represented in the Cyberpunk genre). This idea can be understood as in opposition to the way some think of how the technology industry and bureaucracy dominates the world today, where an individual is just a number and knowledge of science is devalued. This ideal reinforces the importance of human potential, somewhat similar to the Punk ideal of DIY (Do It Itself).

Among the forms of art influenced by Steampunk are fashion, video games, including Thief and Bioshock, comics, such as The League of Extraordinary Gentlemen, and animations like 9. In the context of Japanese anime, the list is huge, including Steamboy, Full Metal Alchemist and Last Exile.

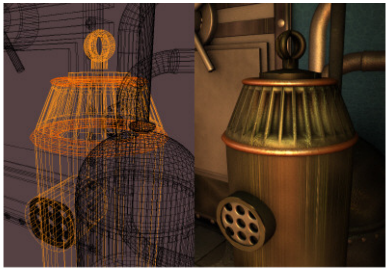
The Steam Side

To give an antiquated look to the machinery, the first step was to create mechanical parts: gears, valves, pipes, tanks, coolers, air ducts, chains, bars and a metal door to populate the scene.

The valves were created from a path made in the open source Inkscape vector editor. The gears were constructed from cylinders, with teeth shaped by multiple extrusions.

Luiz Eduardo Borges

Furthermore, it was used with the Subsurf modifier. The links of the chains are torus shapes modified by extrusion. The tubes were generated as Bezier curves and then converted to meshes to add some details.

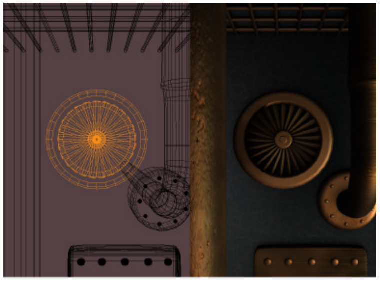


The Punk side

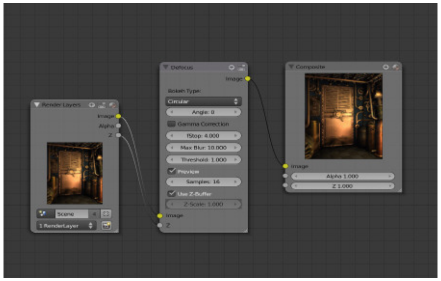
To give a dirty, rusty and irregular look to materials, the textures were composed of bitmaps edited in the image manipulation program GIMP, mixed with procedural textures, influencing both color and intensity of the specular and diffuse reflections, as well as normals.

The material of metal pieces were configured with stronger specular reflection than other elements of the scene. The walls have a bit of gross mirror reflection to give an appearance of moisture. The liquid on the floor uses transparency, reflections and various procedural “Cloud” texture combined on different scales.

Lighting, Composing and Rendering



The light sources of the environment was represented by “Point Lights” in various colors (strong red and yellow, weak green and blue) and a “Sun” to control the atmosphere. Additionally, the “Ambient Occlusion”, “Environment Light” and “Indirect Light” options were also used.



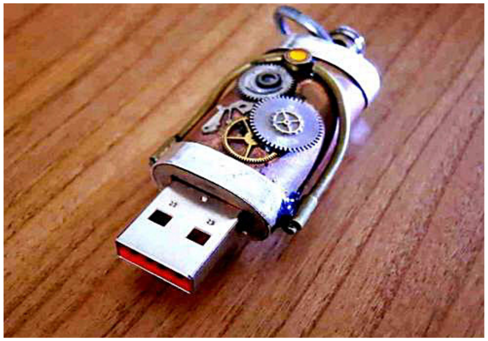
Bokeh Type: Circular
Angle: 0
☐ Gamma Correction
fStop: 4.000
Max Blur: 10.000
Threshold: 1.000
☑ Preview
Samples: 16
☑ Use Z-Buffer
Z-Scale: 1.000
Alpha 1.000
Z 1.000

The “Defocus” node was used to create a small amount of DOF (Depth of Field) effect to simulate the behavior of camera's lens.

The rendering was done using the internal render in Blender 2.52, which tests showed is faster and produced a higher quality output than the production version (2.49).

There are many parts to explore (and model) inside the submarine (it's a work in progress), and a possible next step is a future animation.

3D WORKSHOP: Modeling a Steam Punk USB Flash Drive



USB Flash Drive

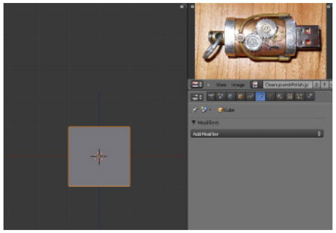
by dreamsgate

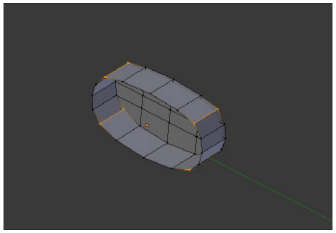
Introduction

While I don't have a lot of experience in mechanical modeling, I thought a simple Steam punk project would be a fun learning experience. So after browsing through a number of images on Google for inspiration, I settled on what looked to be a fairly simple project.

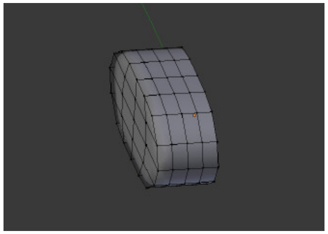
Steam Punk Usb Flash Drive image credits:

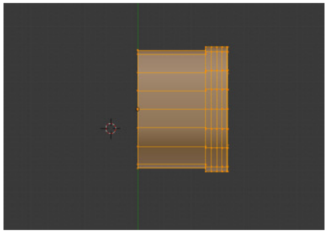
Nathaniel Hursh wrote a great tutorial on Instructables.com on how to make a Steam Punk Usb Flash Drive out of copper pipe, brass tubing and assorted watch parts. He was kind enough to allow me to use his photos of his project for this article.

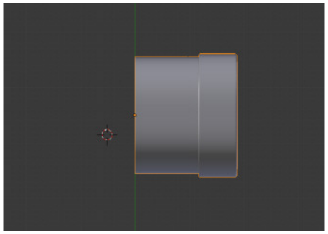


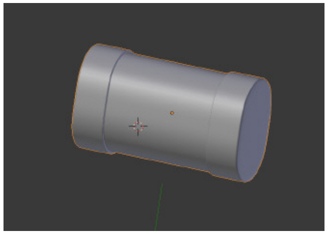


I studied my image for areas that might cause problems and ultimately decided that this was doable. So full of confidence and a fair amount of excitement, I sat down to do a little modeling. Not long into the project, I realized that this wasn't going to lead to an insightful article on modeling a usb drive. In fact the further along I went, the more it started becoming a "Misadventure in Mechanical Modeling".









You would have thought I would have seen this coming, but no I had managed to convince myself that the project was simple enough, that even I couldn't mess it up.

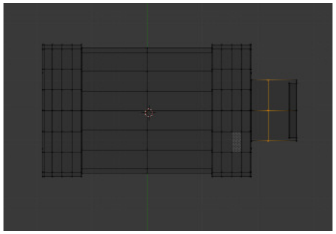
Without fail, every single step led to me trying up to a half dozen approaches to get it to match the image. It should not be this hard, I picked an easy project on purpose. The undo key and I became best friends rather quickly.

Oddly enough the part of the project that I was least concerned about gave me the most problems. You can see from the reference image that the actual body (case) of the usb drive is just a simple oval cylinder.

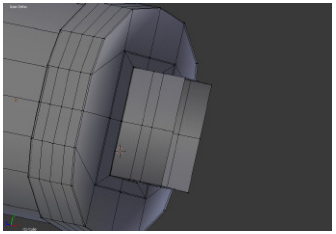
So naturally I added a circle, did a little rescaling to get an oval shape. So far so good. Next up, a little extruding and sizing for the end cap.

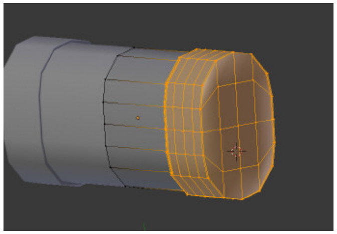
by dreamsgate

Uh oh, I already see a problem, the plug is very rectangular. But the faces on the end of the cap aren't. Well I could just go with it, but this is already starting to feel like it will lead to a major headache.

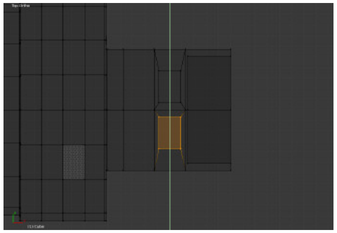


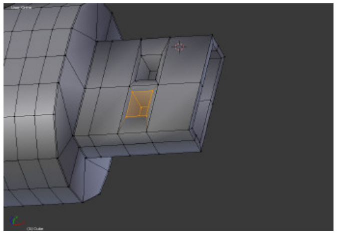
Round 2: okay, time to think smarter. This the time I started with a cube. A little scaling, a few loop cuts and some tweaking of vertices later, I had a fairly nice looking cap shape. Okay now, that's gonna work much better. After extruding out the body of the case and applying a mirror modifier (for the other end), I started the plug.

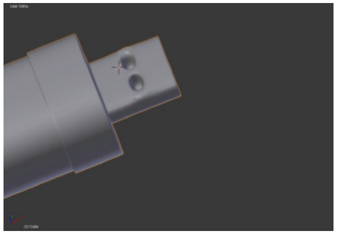




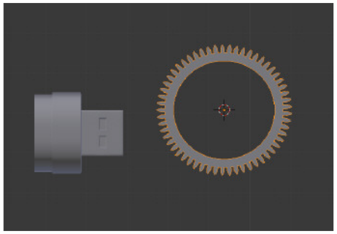
This should be easy as pie, just extrude it out and extrude in some holes. Uh huh, sure... I should have known better. The extrusion part went fine, the hole part, not so much. No matter what I did, the holes insisted on staying round (due to the stupid subsurf modifier which I needed to make the case rounded). I tried varying combinations of loop cuts and creases until I finally got close enough. It works, but it isn't very pretty. And then discovered that I had forgotten to apply the mirror modifier before I modeled the plug. Nice... anyone need a two plugged usb drive? Yeah I didn't think so. So obviously I deleted the extra plug and filled in the faces.

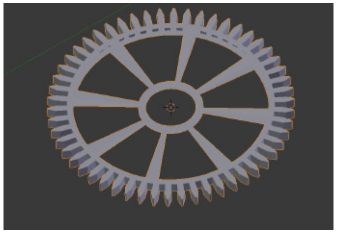






Next up was making a few gears. That part was actually fun. I used the gear add-on script, so it was basically creating them for me. Most of the gears were fairly straightforward, but there were two that required a little bit more detail, in the form of spokes to a center disk. That was actually easy enough to figure out. I just extruded faces toward the center of the gear and then deleted sections to create spokes.

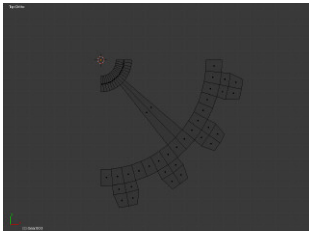


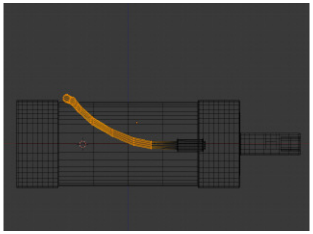


Almost done. Just the tubing to go. I don't know why I thought a simple piece of tubing was going to be easy.

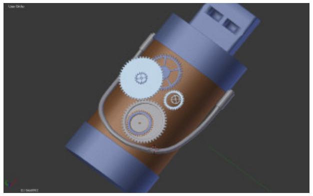
by dreamsgate

Obviously I must have been delirious by this point. My first thought was to create a curve and apply a circle bevel to it, but for some reason my curve just flat out refused to allow me to do that. I tired with both a bezier curve and a path curve. I tried, adding the circle object for bevel in the curve options and as a modifier. I know it is possible, I have done it for non-mechanical objects and models. Sigh... but of course it isn't going to work this time.





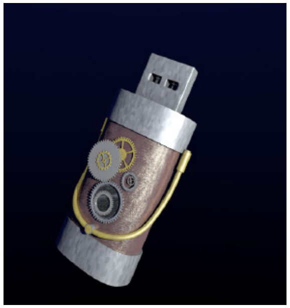
Fine, then extrusion modeling it is. Add a tube and extrude my way up and around the case. Quite a bit of tweaking later, the tube is finally attached. It looks slightly mangled, but there it is.



Finally, a quick stop at the Blender Material Repository for appropriate metal materials and time to render.

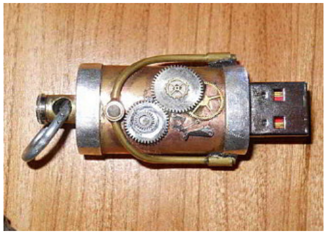
Conclusion:

Even after all my misadventures, it did turn out fairly close to the reference image I was using. As long as you don't peek at the underlying geometry, you might even think I did a good job. The materials and lighting could of course use some more work to make it look even closer to the reference image. But all said and done, this obviously ended up being more of a learning experience for me than for you. In fact I'm betting quite a few of you are desperately trying to hold back your giggles and laughter until I wander off.

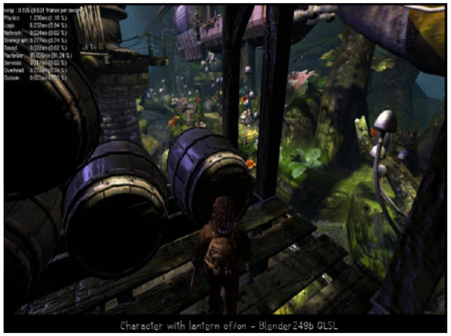


I do believe I have sufficiently tortured myself for the time being, so I am going to go model something with no hard edges, holes, tubes or gears.

bye now....



MAKING OF: Dark Biome



Character with lantern of/on - Blender249b GLSL

Dark Biome

apostroph3d

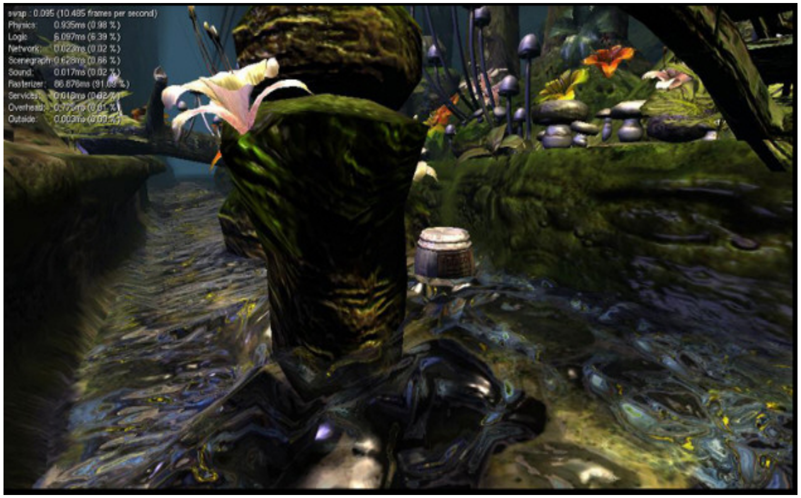
For a long time, I needed a lamp with drop shadows for a 360 degree lantern for the functional character of the games. There is no black market, my NASA programmer buddies are too absorbed in the new space shuttle, and maybe Ton and his team of programmers do not have all day for management of lights in the game engine.

Being too busy with the messianic series Blender 2.5 alpha, and the fantastic Sintel that everyone expects.

Foaming at the mouth. (Quick question, Is yor having a small full nude scene on the dvd?).

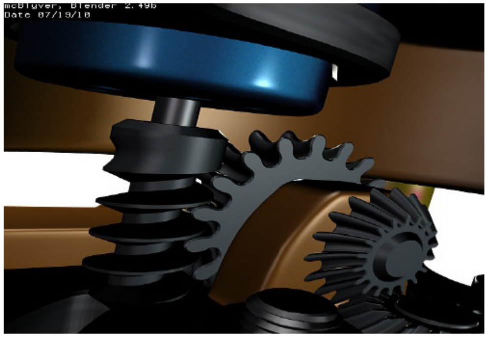
So what to do? After reworking the shape and texture of the lantern. I used a spot with a large opening, the character thus highlights the environment in a single direction.

Watch trailers some here http://www.youtube.com/user/apostroph3d?gl=FR&hl=fr.



```
swap : 0.095 (10.485 frames per second)
Physics:     0.935ms (0.98 %)
Logic:       6.097ms (6.39 %)
Network:     0.022ms (0.02 %)
Scenegraph:  0.628ms (0.66 %)
Sound:       0.017ms (0.02 %)
Rasterizer:  86.876ms (91.08 %)
Services:    0.018ms (0.02 %)
Overhead:    0.775ms (0.01 %)
Outside:     0.003ms (0.00 %)
```

Hey! Ton! when will you include the Alpha Shadows in Blender by default? ...



Advanced Py Drivers

Alain Mathez

I began a project with an industrial robot. The goal was more or less the same as with caterpillar, animate the whole thing with as few Ipo(s) as possible. So naturally the position of the arm and the "hand" are driven with an Empty. But each of the direction (X, Y and Z) of the Empty drives a separate function of the robot.

You can see the hand in action here :

The entrails here :

A short anim here :

And another anim of the hand here :

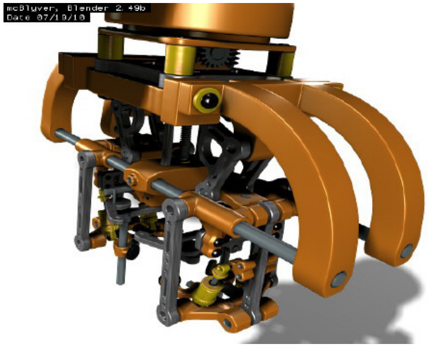
And the "work in progress" here :

Sorry,it's all in french. My problem now, is that I'm far far away from being finished, and I have absolutely no time at the moment to work on it. I am not like Andrew Price who is 100% a BlenderGuru, but I will try the following way ;-) Next week I will co-organise a Blender-Day in Switzerland (for 10 children), details for which are on Blendernation :

Blender is with me day and night, but I can't do everything at the same time, so If you want to help please reach out to the people and let them know that they can follow the work on the french BlenderClan.

I have a lot of mechanical tests, but nothing ready to show. So I am providing you a .blend file to play with, which have something like "Advanced PyDrivers Theme".

The principle is to drive constraints with mathematical formula to decompose the only X movement of the Empty "Master" in defined steps.



With this method I can drive a Geneva Drive in real time, and a mechanical differential with a brake system at the same time, by just adding a "very simple to rig" a connecting rod with 3 Bones and IK (just bonus)

Everything is real time, that means no leg in animation while running it at slow, fast, stop, reverse... with every element perfectly synchronised, and playing with only one lonely IPO (LocX).

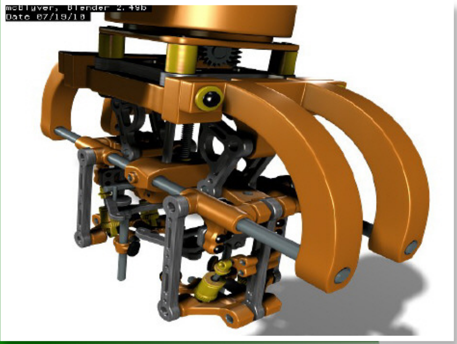
Blender file is not configured to render a "wow" image or animation, it is just ready to play with Empty Master and observe what happens.

Visuals objects are on layer 11-14, and Armatures and Empties on 1-4.

info@mcblyver.ch

www.mcblyver.ch

ARTICLE: Alternative Realities: Steampunk and Dieselpunk



Alternative Realities: Steampunk & Diesel punk

Meino

Massive smoke, spit towards the reddish sky by hundreds of fires, darkens what was formerly known as daylight. The air smells like a mixture of heated machine oil and metal dust. Huge machinery, covered with rust and the dirt of hard work, whose beginnings are no longer recalled, roar in united rebellion as their hot breath immensely heats up the heavy air.

Welcome to the machine -- welcome to the Steampunk!

The technology of Steampunk originates in steam engines and fictional history that gets stuck somewhere in Victorian time between 1837 and 1901. Steampunk is a subculture, an invention of an alternative reality, which gives the steam engine a central role. Not only are cars and aircraft driven by steam, but also analogous computers and other aspects of modern technology which were reinvented to be steam-based and their style mapped back into the late 19th century. Steampunk not only has its expression in an alternative technology evolution but also heavily influences fashion. Typical accessories of Steampunks are for example welding goggles, belts and things built from cogs. Modern gadgets also become ``steampunked'' by adding cogs, small pipes and changing plastic housings to brass. And of course, no polished shiny brass!

Everything has a used or even a little dirty look. The ``practical aspect'' of all things is everywhere. And by all means: Do not forget the goggles! Goggles are that important! Steampunk can be seen as a form of science fiction which assumes that the industrial revolution has taken an alternative path in time and steam engines were the one and only source of power. Often Steampunk presents a dystopian aspect. Another kind of alternative reality is Dieselpunk which uses a comparable setup but the technological assumption here is that diesel engines replace the steam engines used in the world of Steampunk. The distinction from Steampunk isn't that sharp. Often Dieselpunk is even more dystopian the assumptions for this alternative reality do not exclude a doomsday which has taken place somewhere in the past and mankind has been thrown back to the technological achievements of the diesel machinery era. As in the Steampunk reality, modern technique is back-ported to be ``diesel compatible''. Both Steampunk and Dieselpunk often include a kind of retro-science fiction. Only one word: Rayguns! By all means: Rayguns!!!

Retro-science fiction comes in two forms: One is the Buck Rogers and Flash Gordon type. It is simply the science fiction of yesteryear.

The second one is science fiction of today back-ported into the fictional world of Dieselpunk and Steampunk.

Both alternative realities Steampunk and Dieselpunk are fictional but in their dystopian aspects may reflect the threat and angst we feel toward the problems which modern technology will cause in the future.

A totally different and overwhelmingly optimistic prediction of the future was dared in the era of the Googie design however and it was thought to be everything but fictional. It was believed to become true.

Googie, also known as ``Populuxe'' or ``Doo-Wop'', was an architectural design which started in California in about the 1940's and ended in the late 1960's.

Meino

But it also influenced other aspects of daily life. It was the time of the beginning of the car culture in America when everything seemed possible. Secret time-machines which looked like two-transistor radios? Only a matter of design! Flying cars? Why not ??! Look at those fins of the automobiles of that time! Holidays on the moon? Absolutely! Trips to outer space with a single-stage rocket? It was not easier than that (why do you ask?)!

It was also the time of the rayguns, incredible monsters from behind the stars and a very simplistic assumption: beings are either good or damn evil. Especially foreign beings -- aliens -- were damn evil. So folks, enough theory -- fire up your steam engines! The generators are thirsting to feed their electricity into your Blender 3D-generators, which have waited so long to make these dreams become true. Because a picture can say more than a thousand words a list of links follows. The list is of no specific order and by no means asserts the claim of being complete or representative. The links point to sites which have more information about Steampunk, Dieselpunk and the Googie design. Though the author has taken care what to link here,

Steampunk sites:

- http://en.wikipedia.org/wiki/Steampunk
- http://www.nytimes.com/
- http://www.woodysnet.co.uk/steampunk
- http://www.steampunk.republika.pl/main.html
- http://brassgoggles.co.uk/blog/
- http://www.pinkraygun.com/
- http://etheremporium.pbworks.com/What+is+Steampunk
- http://etheremporium.pbworks.com/Neverwas-Haul
- http://steampunkworkshop.com/
- http://www.datamancer.net/
- http://www.philm.demon.co.uk/Steampunk/
- http://www.theideacan.com/
- http://thesteamblog.wordpress.com/
- http://antiquescollectibles.suite101.com/
- http://www.suddite101.com/
- http://www.suite101.com/reference/steampunk
- http://www.suite101.com/reference/steam_punk
- http://www.suite101.com/reference/steampunk_design
- http://www.suite101.com/reference/victorian_style
- http://daily-steampunk.com/
- http://steampunkpics.blogspot.com/
- http://steampunkpics.blogspot.com/2008/11/weapons.html
- http://steampunk-creator.deviantart.com/
- http://www.meetup.com/steamhub/
- http://hacknmod.com/

Dieselpunk - sites:

- http://en.wikipedia.org/wiki/User:Piecraft/Dieselpunk
- http://en.wikipedia.org/wiki/Cyberpunk_derivatives
- http://www.omnipelagos.com/entry?n=dieselpunk
- http://www.dieselpunks.org/
- http://dieselpunks.blogspot.com/
- http://www.darkroastedblend.com/2008/12/dieselpunk

- http://dieselpunk.net/
- http://en.wikipedia.org/wiki/User:Piecraft/Dieselpunk
- http://www.ottens.co.uk/gatehouse/dieselpunk.php
- http://www.cutoutandkeep.net/projects/dieselpunk
- ttp://www.chrisroberson.net/2008/07/dieselpunk.html
- http://www.ottens.co.uk/gatehouse/gazette-3...
- http://www.ottens.co.uk/gatehouse/gazette.php
- http://dieselpunkpics.blogspot.com/

Films/movies:

“Brazil”, “Sky Captain and the world of tomorrow”, “Matrix”, “Dark City”, “Metropolis’ ...and even more *punks are here to discover:

- http://blastr.com/2009/09/guide-to-sci-fi-punks.php
- http://antiquescollectibles.suite101.com/article.cfm/

Googie sites

- http://en.wikipedia.org/wiki/Retro-futurism
- http://en.wikipedia.org/wiki/Googie
- http://www.retroplanet.com/blog/retro-design/
- http://silver-rockets.com/
- http://www.plan59.com/main.htm
- http://www.seattledreamhomes.com/PageManager/
- http://www.superaardvark.com/?p=140
- http://www.lostdiscsradio.com/
- http://www.darkroastedblend.com/2007/11/
- http://weburbanist.com/2009/03/02/retrofuture
- http://www.radiosalon.com/html/radios1.htm
- http://www.ultraswank.net/
- http://gajitz.com/the-future-of-the-past-16
- http://shop.webomator.com/
- http://gadgets.boingboing.net/retrofuturism/
- http://www.digitaldeliftp.com/Recommendations/
- http://fiveprime.org/hivemind/Tags/design,googie
- http://www.jamesbutters.com/
- http://www.jamesbutters.com/sharpbh351rocket
- http://www.drive-on-in.com/googie.htm
- http://www.retromatic.tv/cpg1410/displayimage
- http://community.livejournal.com/googielove
- http://www.smashingmagazine.com/2008/04/21/
- http://hacknmod.com/hack/epic-collection-of-sci-fi-ray...

These links do not directly point to steampunk, dieselpunk or Googie design but they share some aspects of them for your inspiration: These two sites also partly express the more post-doomsday and apocalyptic character of steampunk and especially dieselpunk in a very direct way.

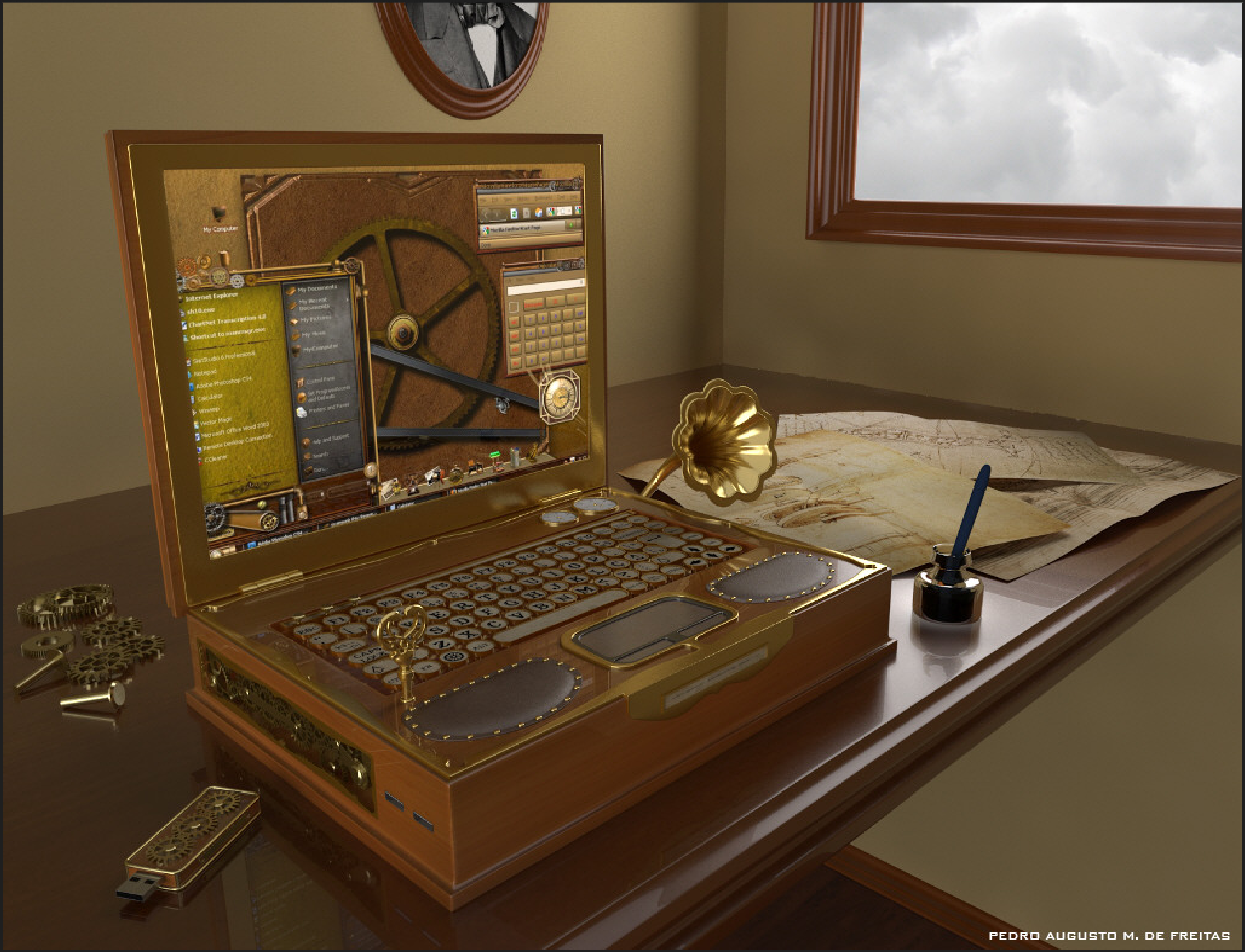
- http://www.massiveblack.com/MBwebolution/
- http://conceptships.blogspot.com/
- http://fusionfilter.com/
- http://dqsketches.blogspot.com/

Have a lot of fun!

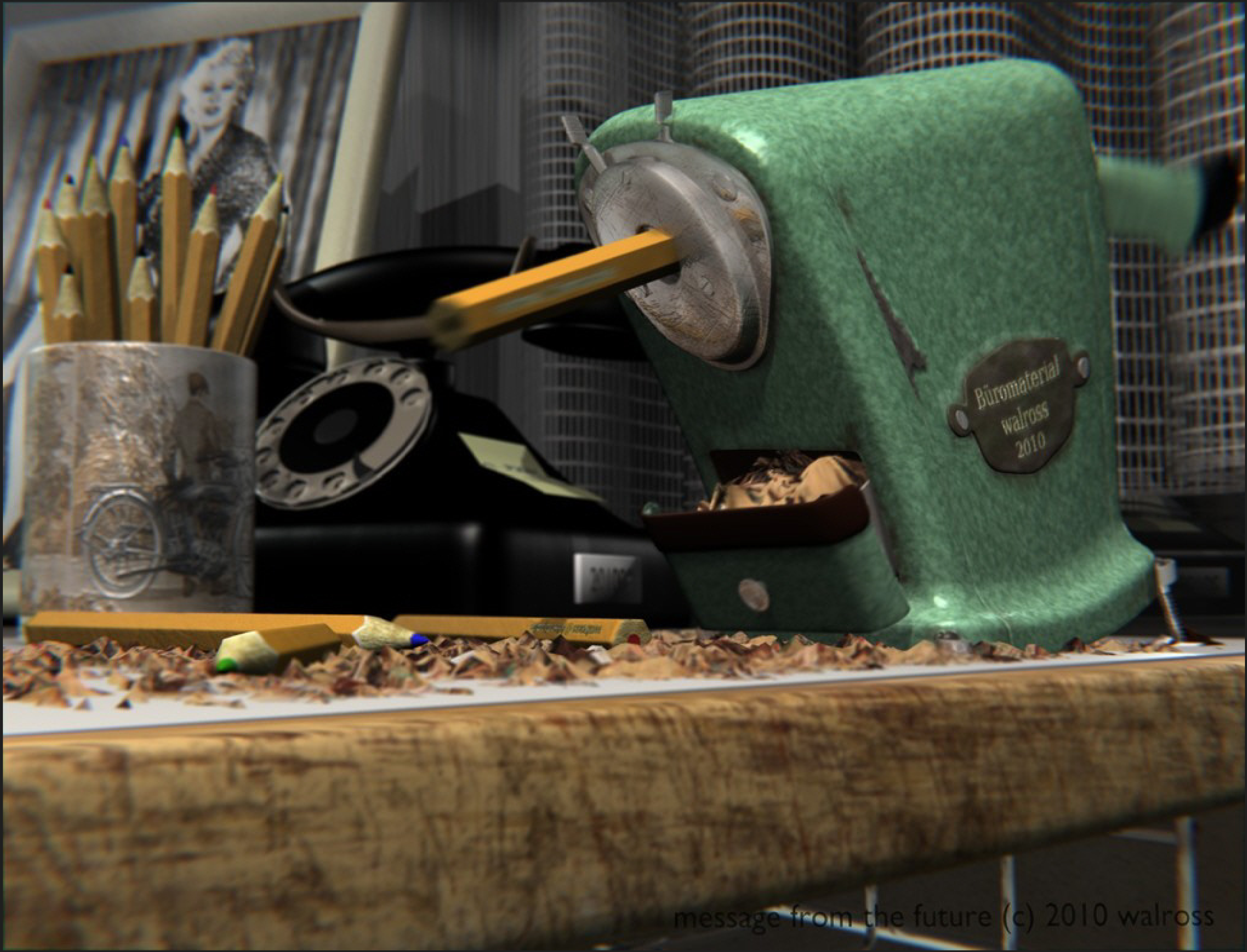
Meino

Meino

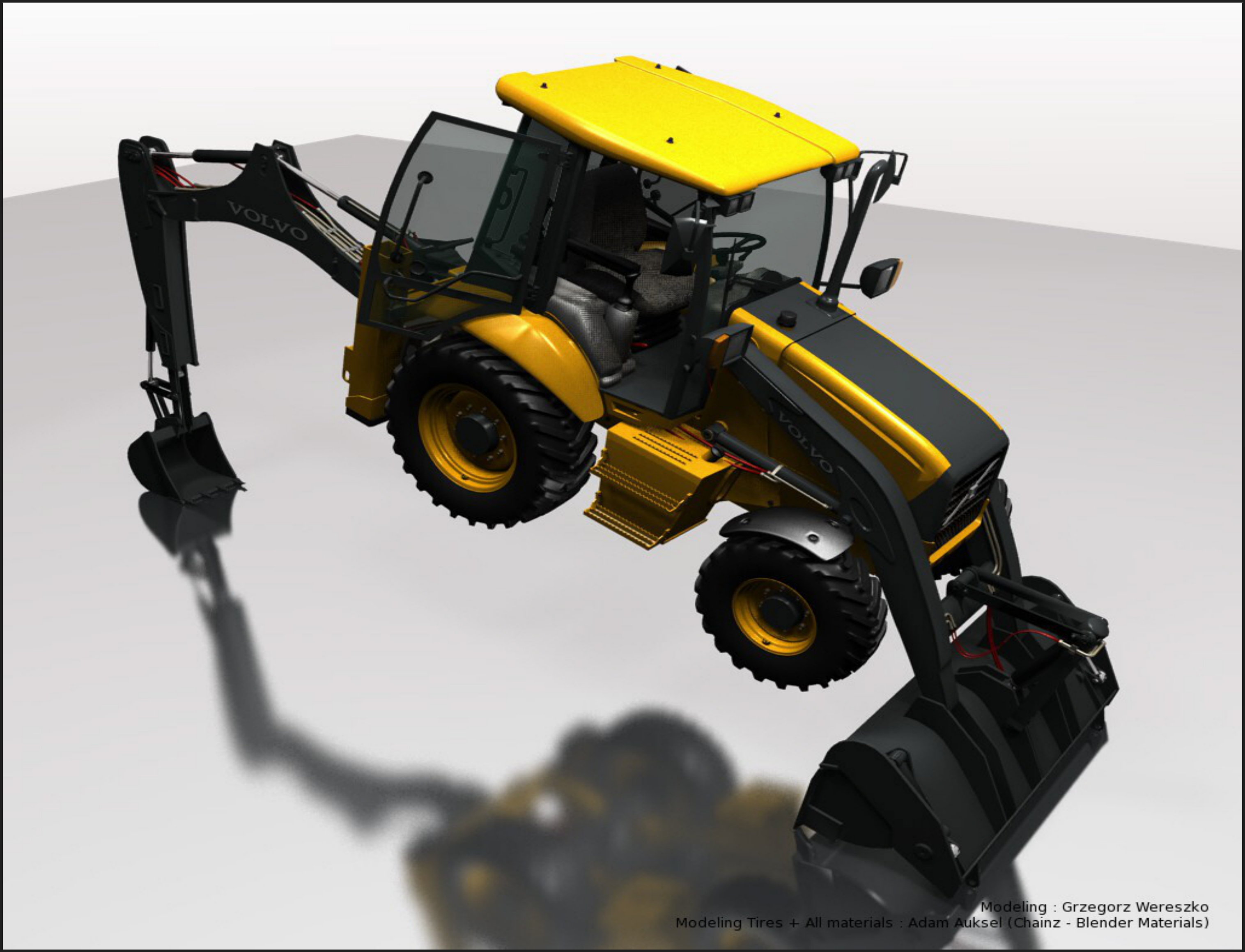
The "Old" Notebook - By Pedro Augusto



Message from the future - by Hans Schwaiger



Loader

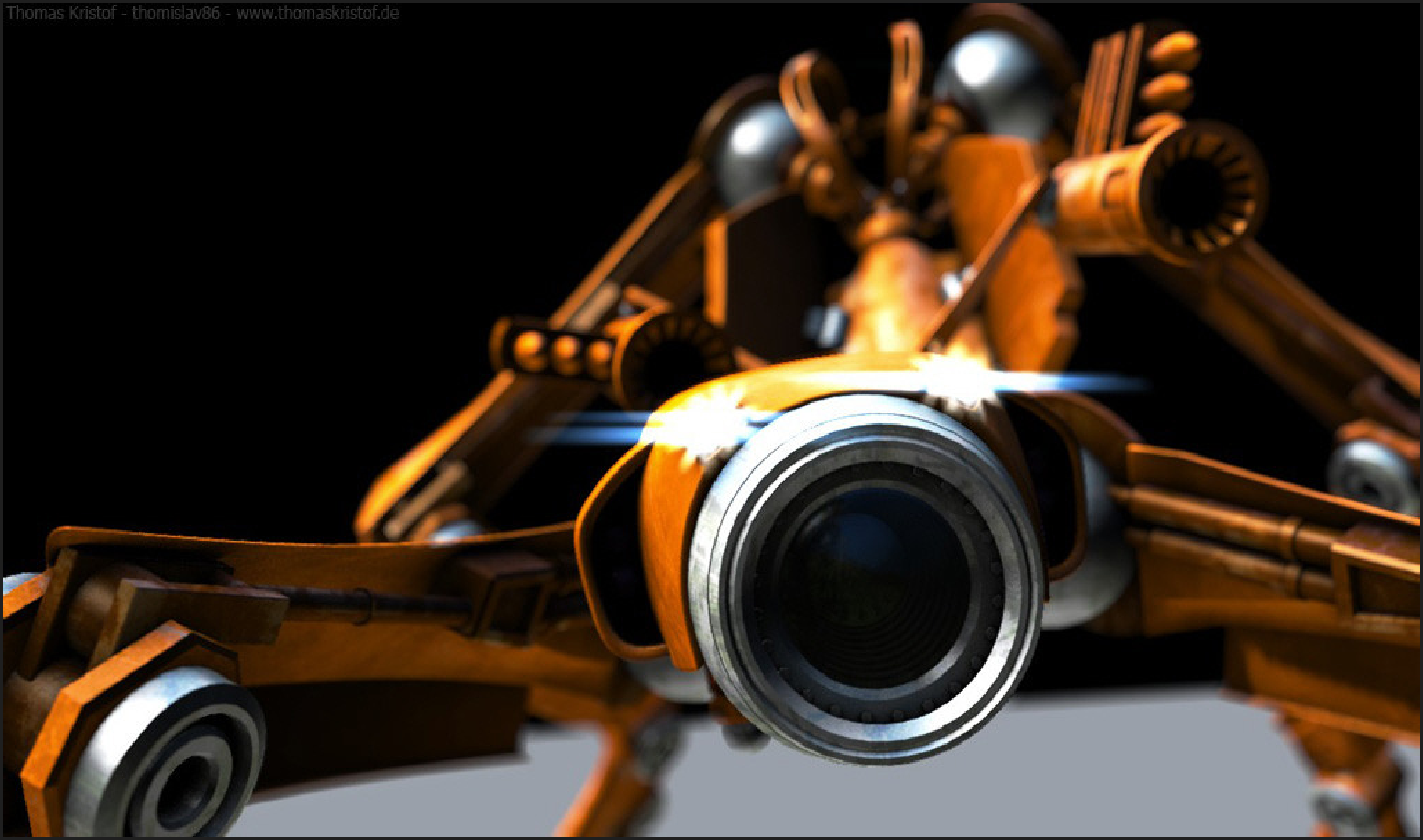


Modeling : Grzegorz Wereszko
Modeling Tires + All materials : Adam Auksel (Chainz - Blender Materials)

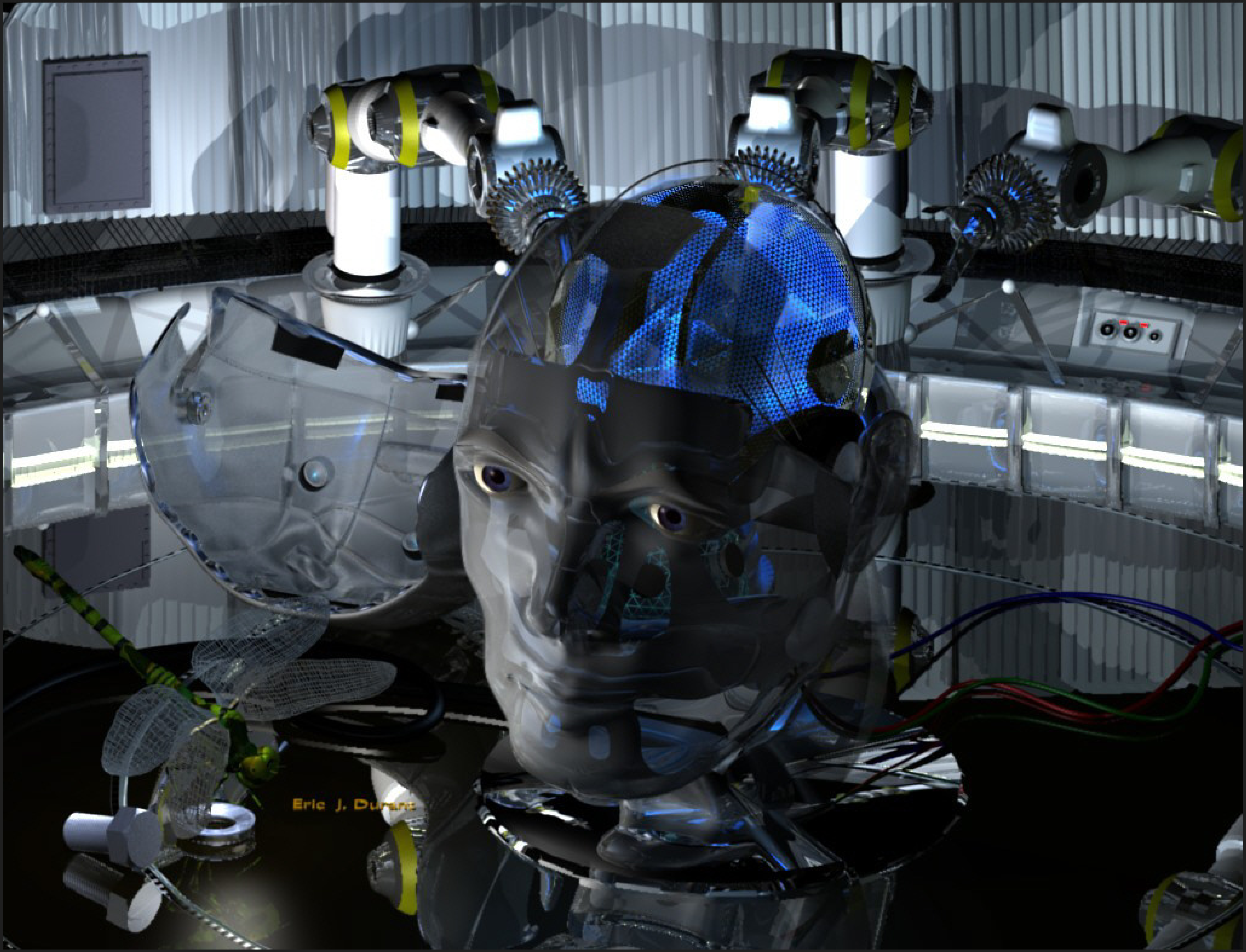
Loader

Modeling : Grzegorz Wereszko
Modeling Tires + All materials : Adam Auksel (Chainz - Blender Materials)

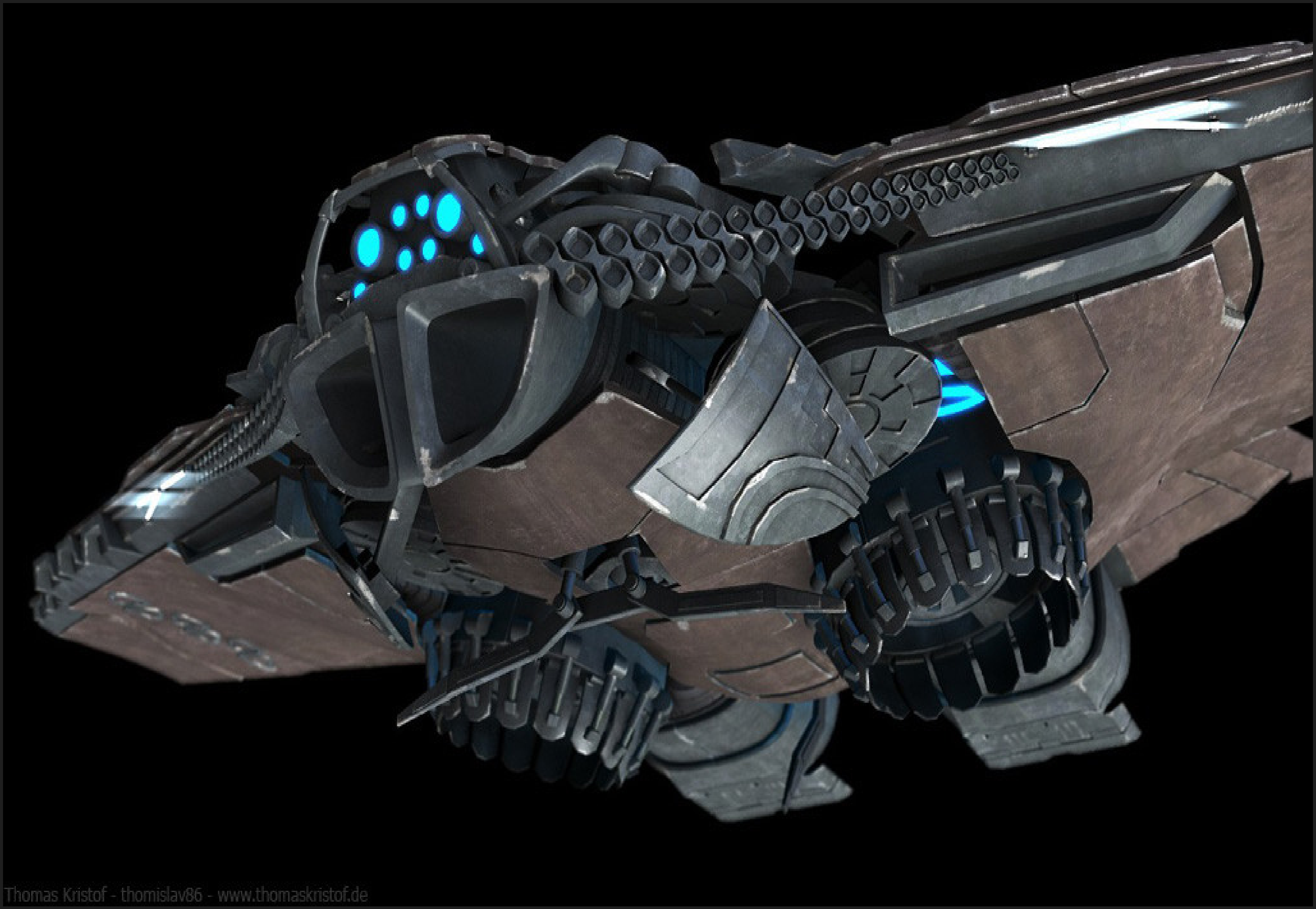
AMEE - by Thomas Kristof



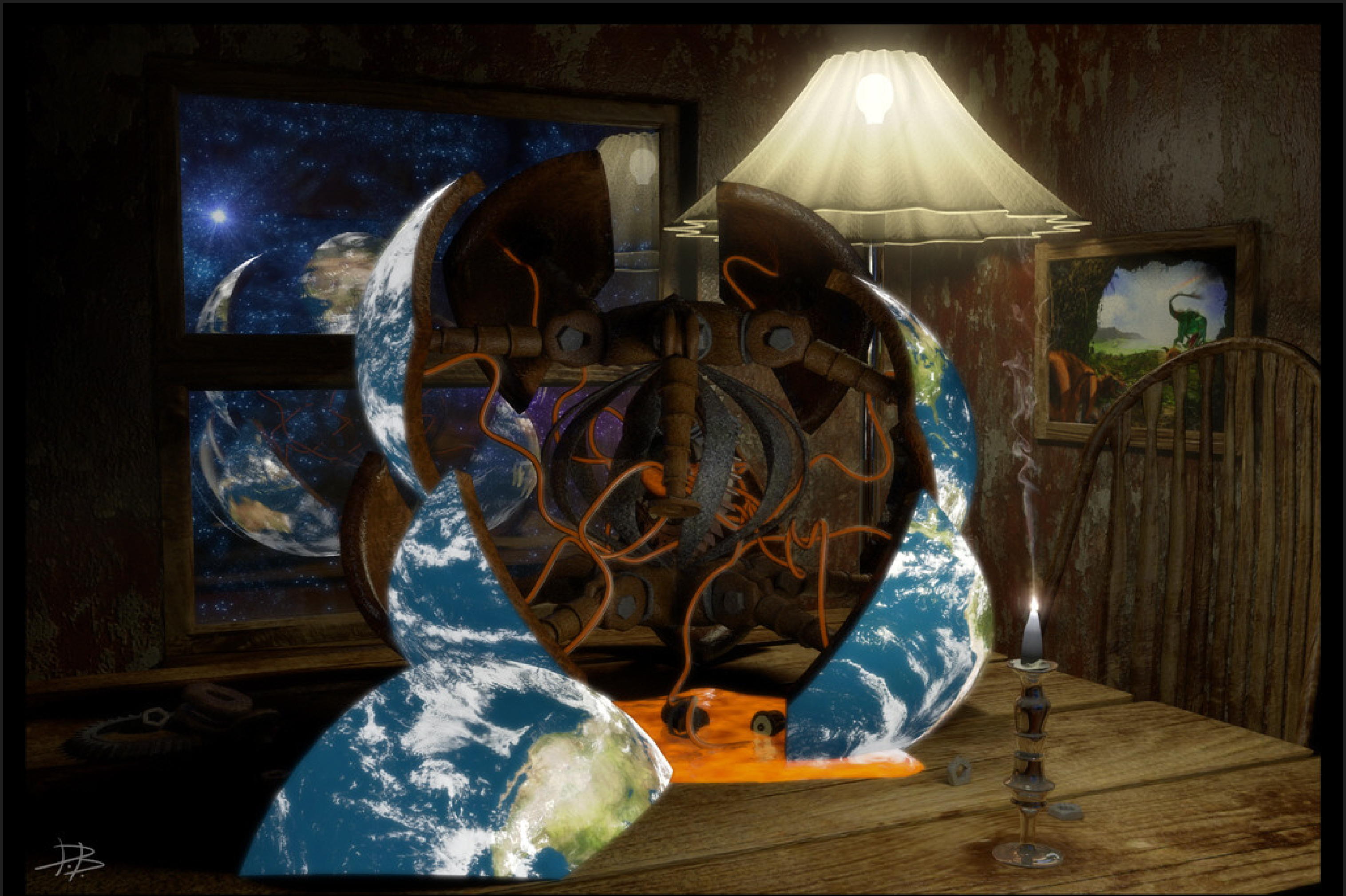
Assembly Required - by Eric Durant



Gleiter - by Thomas Kristof

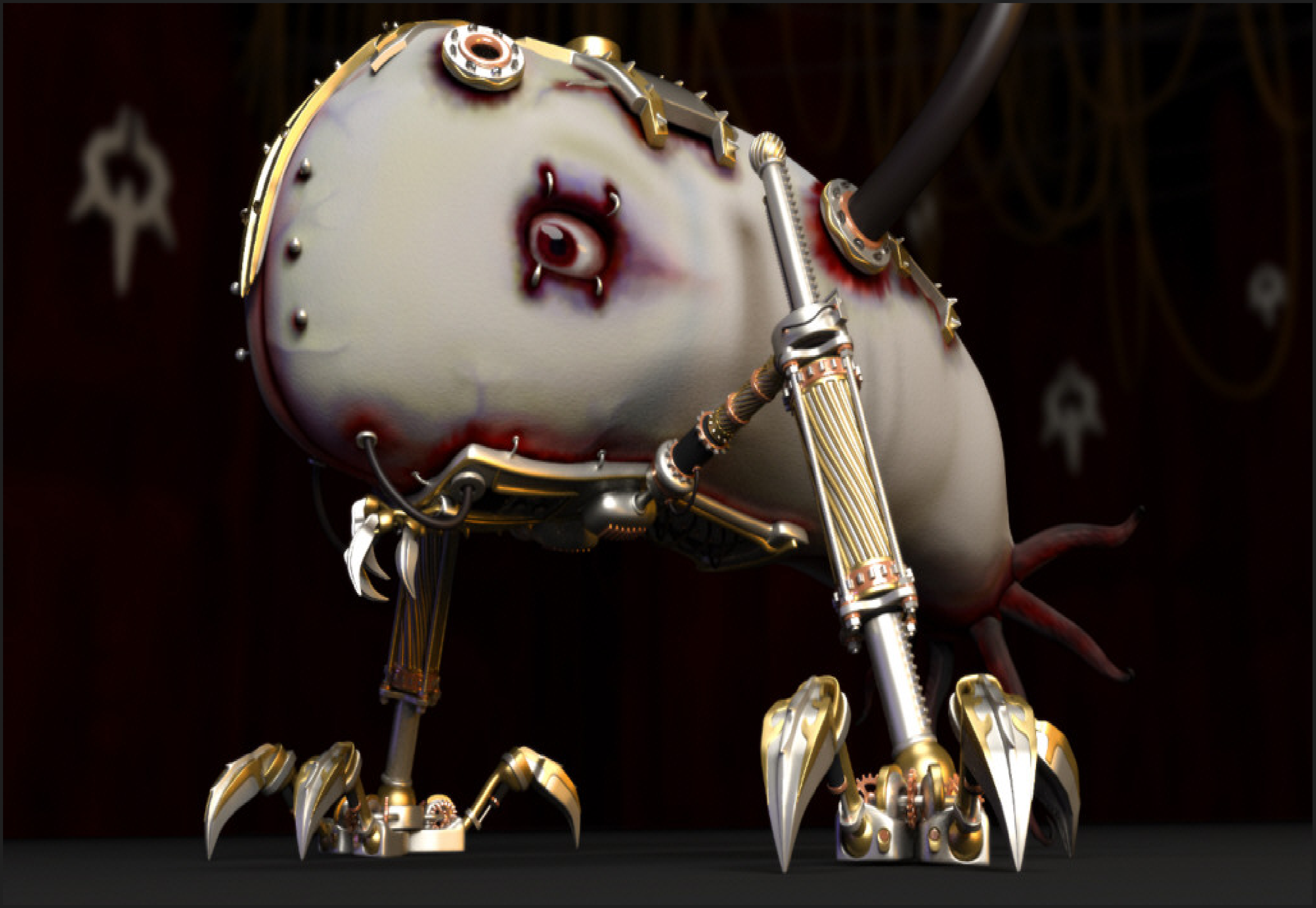


OutOfWarranty - by Daniel Brown

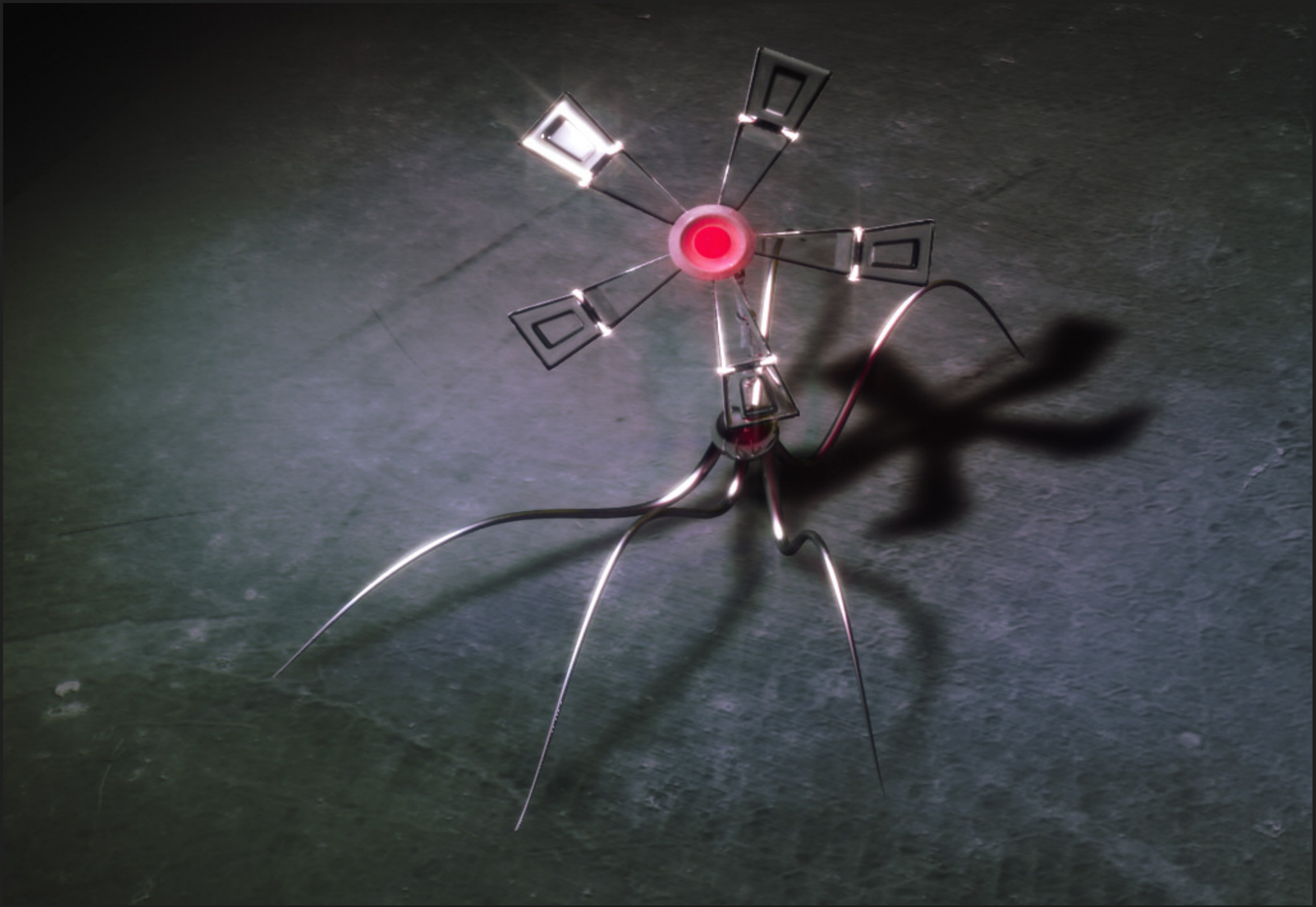


"Out of Warranty" • Daniel DeWitt Brown © 2010

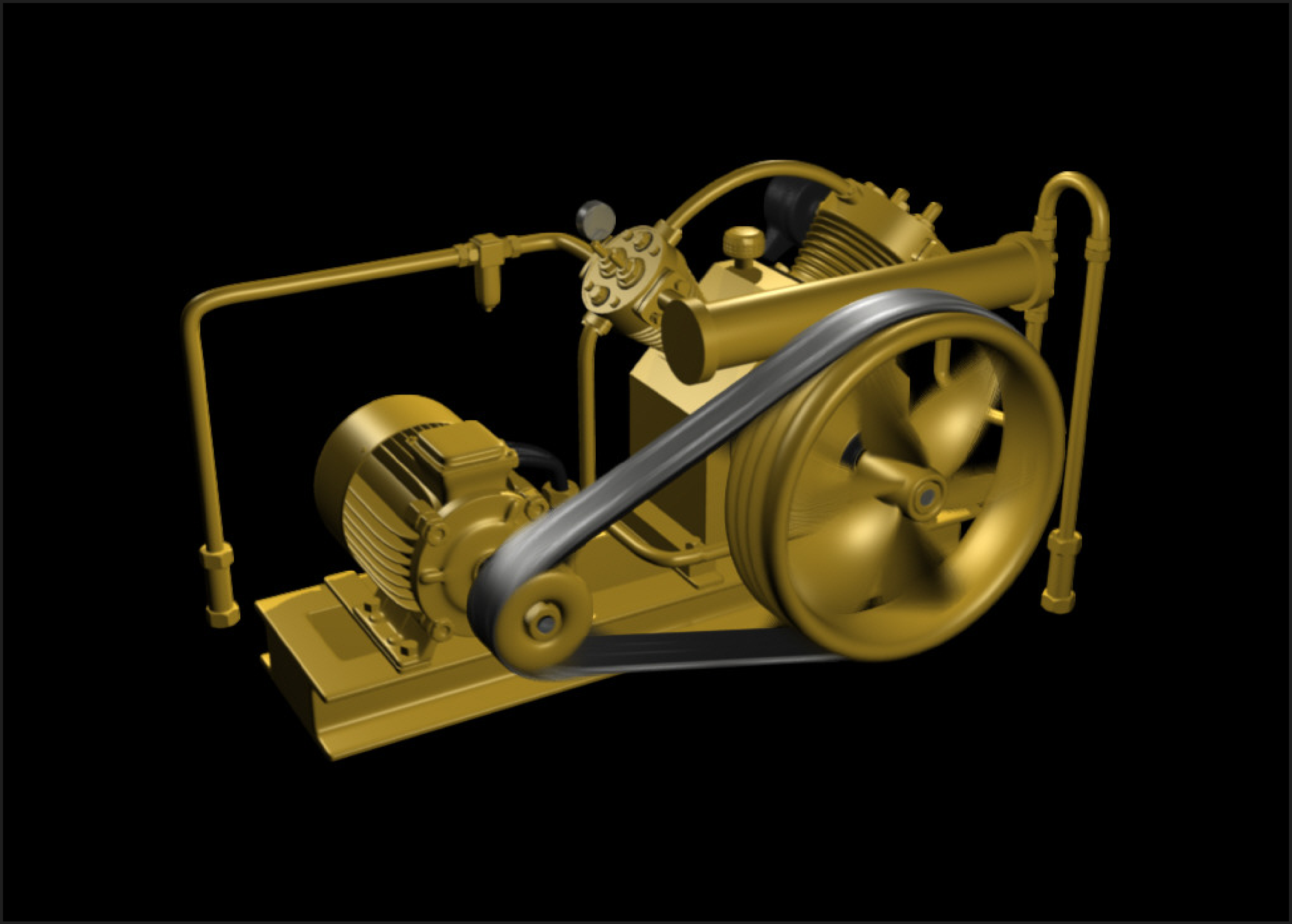
Steampunk strogg - by Vadim Andreyev aka sid350



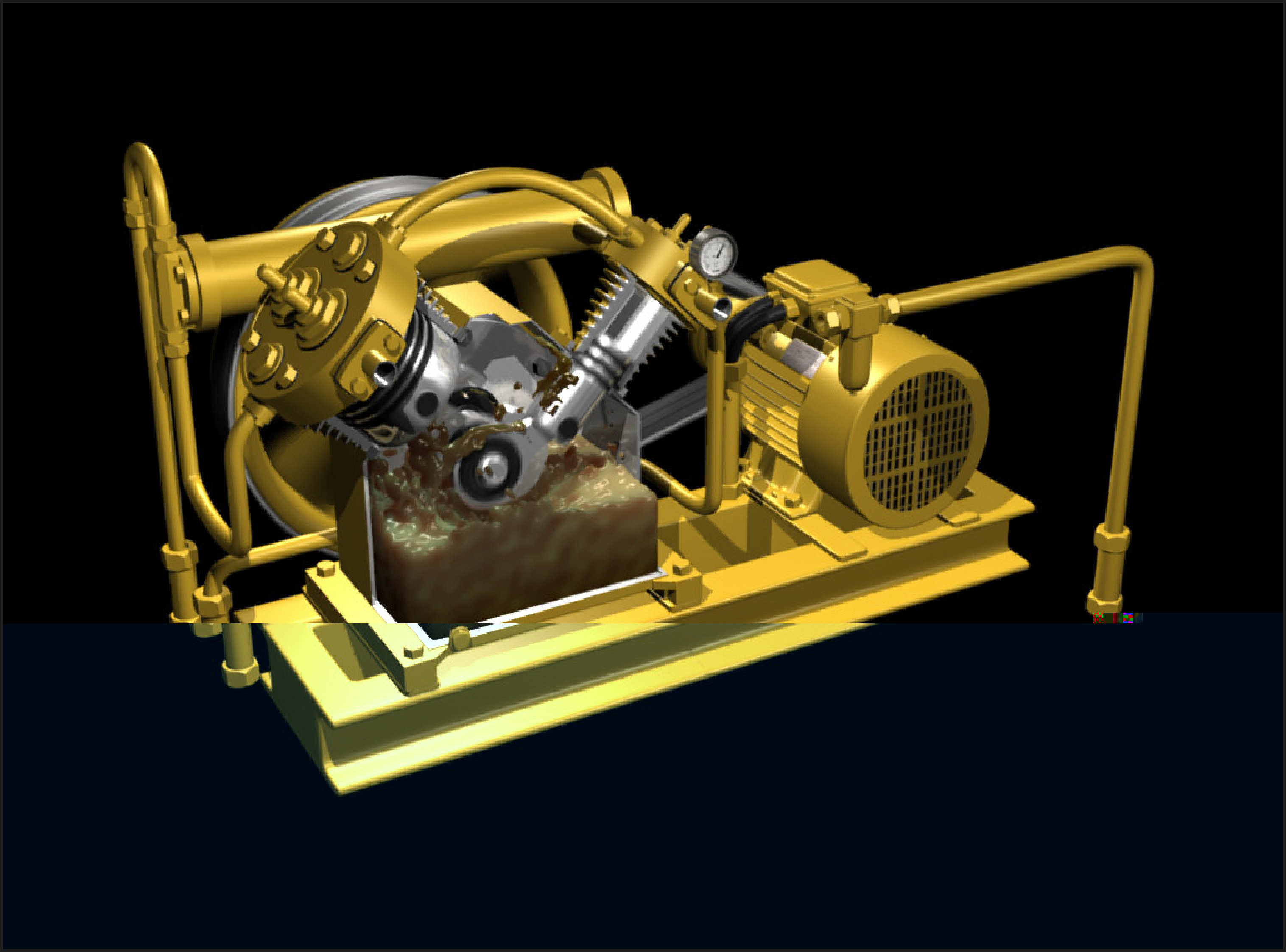
The robot flower_by Yunior Frómeta



ABC Compressor - by Martin Caston



ABC Compressor - by Martin Caston



Want to write for BlenderArt Magazine?

Here is how!

1. We accept the following:

- Tutorials explaining new Blender features, 3dconcepts, techniques or articles based on current theme of the magazine.
- Reports on useful Blender events throughout the world.
- Cartoons related to blender world.

2. Send submissions to sandra@blenderart.org. Send us a notification on what you want to write and we can follow up from there. (Some guidelines you must follow)

- Images are preferred in PNG but good quality JPG can also do. Images should be separate from the text document.
- Make sure that screenshots are clear and readable and the renders should be at least 800px, but not more than 1600px at maximum.
- Sequential naming of images like, image 001.png... etc.
- Text should be in either ODT, DOC, TXT or HTML.
- Archive them using 7zip or RAR or less preferably zip.

3. Please include the following in your email:

- Name: This can be your full name or blenderartist avtar.
- Photograph: As PNG and maximum width of 256Px. (Only if submitting the article for the first time)
- About yourself: Max 25 words .
- Website: (optional)

Note: All the approved submissions can be placed in the final issue or subsequent issue if deemed fit. All submissions will be cropped/modified if necessary. For more details see the blenderart website.

Issue 30

"Once upon... an image? the art of one image storytelling”

- You are free to focus on any aspect of your image including:
- lighting
- composition /staging
- materials
- subject matter / modeling

Disclaimer

blenderart.org does not takes any responsibility both expressed or implied for the material and its nature or accuracy of the information which is published in this PDF magazine. All the materials presented in this PDF magazine have been produced with the expressed permission of their respective authors/owners. blenderart.org and the contributors disclaim all warranties, expressed or implied, including, but not limited to implied warranties of merchantability or fitness for a particular purpose. All images and materials present in this document are printed/re-printed with expressed permission from the authors/owners.

This PDF magazine is archived and available from the blenderart.org website. The blenderart magazine is made available under Creative Commons‘ Attribution-NoDerivs 2.5’ license.